

Virtual Mass Technique for Computing Space Trajectories

Final Report

FACILITY FORM 602

N66-19032	
(ACCESSION NUMBER)	(THRU)
<u>164</u>	<u>1</u>
(PAGES)	(CODE)
<u>CR 65227</u>	<u>30</u>
(NASA CR OR TMX OR AD NUMBER)	(CATEGORY)

GPO PRICE \$ _____

CFSTI PRICE(S) \$ _____

Hard copy (HC) to 400

Microfiche (MF) .75

653 July 65

MARTIN

Virtual Mass Technique for Computing Space Trajectories

Final Report

Contract No. NAS 9-4370

ER 14045

January 1966

by D. H. Novak

MARTIN
BALTIMORE DIVISION
BALTIMORE, MARYLAND 21203

FOREWORD

The work described in this report was performed by the Martin Company for the NASA Manned Spacecraft Center under Contract No. NAS 9-4370.

CONTENTS

	Page
Foreword	iii
Summary	vii
I. Introduction	1
II. Basic Principles of the Virtual Mass	3
A. Review of the Restricted Three-Body Problem in Terms of the Gravisphere	3
B. Generalization of the Gravispheric Force Center Concept to the Case of More Than Two Gravi- tating Bodies	6
C. The Solution of the N-Body Problem as Viewed in the Light of the Virtual Mass	10
III. Digital Computation Formulation Considerations . . .	15
A. Vector Orbital Elements	15
B. Noniterated Versus Iterated Computation	22
C. The Computing Interval	24
IV. Digital Computer Study	29
A. Approach	29
B. Results	30
V. General Description and Use of the Virtual Mass Pro- gram for Computing Space Trajectories	41
A. General Description	41
B. Input and Output	43
C. Sample Problem	46

CONTENTS (continued)

	Page
VI. Details of the Computer Program	67
A. Flow Diagrams	67
B. Array Notation	71
C. FORTRAN IV Listing of Program	73
VII. Conclusions and Recommendations	91
A. Conclusions	91
B. Recommended Further Studies	91
VIII. References	93

SUMMARY

19032

This study has demonstrated the feasibility of the Virtual Mass Technique for computing space trajectories and has developed a FORTRAN IV digital computer program for solving the restricted three-body problem by this procedure. The virtual mass at any instant of time replaces the combined gravitational effects of all the real celestial bodies upon a spacecraft. The magnitude and location of this fictitious body, along the line of the instantaneous resultant force vector, are uniquely computed by formulas derived from the generalization of the gravispheric force center concept. The computational procedure is based upon the assumption that, over a small time interval, the spacecraft motion can be represented as a two-body conic section arc relative to the moving and varying virtual mass. In this manner the complete trajectory is computed as a series of such arcs, pieced together in a stepwise manner--updating the position and magnitude of the apparent force center at each step. Thus, the virtual mass technique is like the patched conic approximation in that no differential equations are integrated numerically. It is similar to the Cowell method in that the equations for the virtual mass are much like the acceleration contribution terms in the differential equations of motion. As the spacecraft nears one of the real physical bodies, those terms dominate the contributions of the other bodies and the effective force center approximates that real body in size and location. Finally, this technique displays a kinship to the Encke method in the computation of a reference trajectory relative to the dominant body. This dominating body, however, is the continuously moving and varying virtual mass rather than one of the physical bodies. Since the perturbing effects of all bodies are included in the determination of this apparent force center, effectively a perfect rectification is made at each step and there is no need to numerically integrate these perturbations.

A single compact computer program embodying this procedure can be controlled very simply to compute an approximate solution rapidly as a series of relatively few patched conics or a highly accurate trajectory as a large number of such arcs at the expense of proportionately longer computation time. For example, a 70.33-hr insertion-to-pericyynthion circumlunar trajectory was computed (and a large amount of output data were printed) in 160 seconds on an IBM 7094 computer. This trajectory gave the spacecraft position at pericyynthion accurate to within 0.02 naut mi and exhibited a total variation of the Jacobi energy of less than 2 parts out of 7×10^6 .

Author

BLANK PAGE

I. INTRODUCTION

It is well known that there is no closed-form general solution for the trajectory of an infinitesimal spacecraft freely falling in the combined gravity fields of two or more large celestial bodies. Therefore, each case must be solved individually by an approximate numerical procedure. Currently, two alternative procedures are used for finding such solutions; namely, the patched conic approximation technique and the accurate numerical integration of the differential equations of motion.

The patched conic technique makes the simplifying assumption that, while the spacecraft is within the sphere of influence of any gravitating body, the motion is dominated by that large body to the complete exclusion of all others. Since the general solution to the two-body problem is known to be a Keplerian conic section, a crude approximation to the n-body solution can be computed as a series of these preintegrated conic sections, patched together appropriately at the boundaries of the spheres of influence.

The precise numerical solution of the differential equations involves rather laborious step-by-step computation procedures, based upon one of two fundamental approaches.

The straightforward method of Cowell treats all terms in the differential equations as contributors of equal importance. Most of the time, however, the acceleration experienced by the infinitesimal body is dominated by a single one of the gravitating bodies, and all other contributions are small by comparison. This requires that great care be exercised when combining all the terms so as not to lose the significance of the small contributions. This computational difficulty tends to offset the advantage of the formulational simplicity of this method.

The other basic approach (due to Encke) consists of recognizing this domination of the motion by one body and computing the trajectory in two stages. First, the position and velocity at some time (epoch) are considered to define the elements of the osculating Keplerian conic section relative to the dominant body. Then, the perturbing relative accelerations of the less influential bodies are numerically integrated, carrying comparatively few significant digits, to obtain the path correction to be applied to the basic Keplerian motion. As the magnitude of the perturbed motion grows larger, accuracy would be lost without carrying more significant figures. Instead, when this happens, the reference conic section is "rectified" to obtain new osculating elements at a later epoch, thus reducing the magnitude of the perturbations.

This procedure works well for the case where the motion is always dominated by one particular body, as is the case for the planetary motions about the sun. Spacecraft on lunar or interplanetary trajectories,

on the other hand, traverse from one sphere of influence to another--falling under the domination of successively different reference bodies. During the transitions from one reference body to another, the spacecraft is literally torn between the two major attractions. Sophisticated logic is required to enable the computer to select the dominant body and to switch from one reference body to another to ensure that this computational discontinuity does not disturb the continuity of the trajectory being integrated.

This sophistication is generally considered worthwhile, for the Encke integration step size can be much larger than that of the Cowell method. The final selection of one or the other probably is more a matter of personal preference, however, since the Cowell step can be executed much faster.

Regardless of which integration procedure is used, the solution methods described offer the choice between a very crude, rapidly computed, patched conic trajectory and a high precision comparatively slow-running integrated solution. The former type is useful for parametric studies and early mission planning purposes to determine approximate injection conditions. The latter is needed for the refinement of rough initial conditions into an accurate determination of the requirements for a specific mission. Aside from the fact that two different programs are needed, this refinement process may involve iterative computation with the accurate but slow-running program. This is due to the wide gap between the crude approximation and the precision solution and to the very high sensitivity of space trajectories to errors in initial conditions.

This report describes a unique method of computing n-body trajectories which offers, in a single digital computer program, the capability of efficiently covering the complete spectrum from rapid crude solutions to more time-consuming accurate solutions. Chapter II describes the basic concept upon which the computation is based, and Chapter III discusses various considerations which must be made in mechanizing this concept for digital computation. Chapter IV presents the quantitative results of the study of these considerations, thereby showing how these items have been implemented. Chapter V gives a general description of the computer program and complete instructions in the use of it. For the reader who is interested or who desires to make changes for his own requirements, a detailed description is given in Chapter VI, including a complete FORTRAN listing of the program.

II. BASIC PRINCIPLES OF THE VIRTUAL MASS

The concept of the virtual mass is based upon the idea of replacing the combined gravitational effects of many large celestial bodies upon an infinitesimal spacecraft by the attraction of a single equivalent body. This fundamental idea is not new. Its natural applicability to the restricted three-body problem (two large masses and one infinitesimal mass) is described in Refs. 1 and 2. A rather arbitrary attempt was made to make a similar reduction of the n -body problem in Ref. 3. The latter consisted of singling one point out of the infinite number of possibilities along the line of the instantaneous resultant gravitational force on the vehicle. Once the location (assumed inertially fixed) was chosen, of course, the mass magnitude was determined to give the correct force. The virtual mass location and magnitude, described in this report however, are derived as the n -body generalization of the gravispheric force center associated with the restricted three-body problem. Therefore, the presentation begins with a brief review of what is already known about the restricted three-body problem and proceeds from there with the generalization to the case of more than two gravitating bodies.

A. REVIEW OF THE RESTRICTED THREE-BODY PROBLEM IN TERMS OF THE GRAVISPHERE

Consider the simple system comprised of only two large magnitude point masses μ_1 and μ_2 and (by comparison) an infinitesimal mass spacecraft S . The designation of the mass by the symbol μ is intended to suggest that the real quantity of interest is the mass times the Universal Gravitation Constant. The locus of all spacecraft positions S with constant ratio ρ of distances r_{1S} , r_{2S} to the two masses is a sphere with center G on the line through μ_1 and μ_2 as shown in Fig. 1. Since the gravitational attraction depends only upon displacement from the mass, the ratio of the gravitational attractions is also constant on such a spherical surface; hence, it is called a gravisphere.

The gravisphere exhibits an interesting intrinsic physical property; namely that, for all points on its surface, the resultant \vec{F}_R of the attractions \vec{F}_1 , \vec{F}_2 of the two bodies passes through a single focal point V on the line between μ_1 and μ_2 as shown in Fig. 2. The location of V relative to μ_1 can be shown (e. g. from relations derived in Ref. 2) to be

$$\vec{r}_{v1} = \vec{r}_{21} \frac{\frac{\mu_2}{r_{2s}^3}}{\frac{\mu_1}{r_{1s}^3} + \frac{\mu_2}{r_{2s}^3}}$$

where

$$\begin{aligned}\vec{r}_{ij} &= \vec{r}_i - \vec{r}_j \\ r_{ij} &= |\vec{r}_{ij}|\end{aligned}$$

The location of this gravispheric force center can also be expressed relative to the same frame to which the masses are referred:

$$\vec{r}_v = \vec{r}_1 + \vec{r}_{v1} = \vec{r}_1 + (\vec{r}_2 - \vec{r}_1) \frac{\frac{\mu_2}{r_{2s}^3}}{\frac{\mu_1}{r_{1s}^3} + \frac{\mu_2}{r_{2s}^3}}$$

or

$$\vec{r}_v = \frac{\frac{\mu_1 \vec{r}_1}{r_{1s}^3} + \frac{\mu_2 \vec{r}_2}{r_{2s}^3}}{\frac{\mu_1}{r_{1s}^3} + \frac{\mu_2}{r_{2s}^3}} \quad (\text{II-1})$$

The magnitude of the effective mass (times Universal Gravitation Constant) μ_v which must be concentrated at V to replace the combined effects \vec{F}_R of μ_1 and μ_2 also can be derived from expressions given in Ref. 2 as

$$\mu_v = r_{vs}^3 \left(\frac{\mu_1}{r_{1s}^3} + \frac{\mu_2}{r_{2s}^3} \right) \quad (\text{II-2})$$

Note that, unlike the fixed focal point location, the gravispheric mass magnitude varies according to the radial displacement r_{vs} of the point on the surface from V.

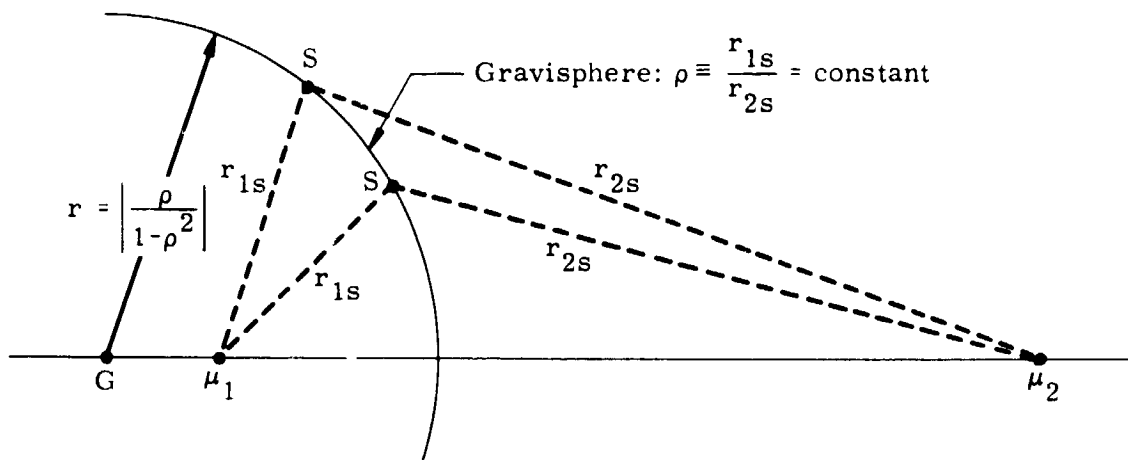


Fig. 1. Illustration of a Gravisphere

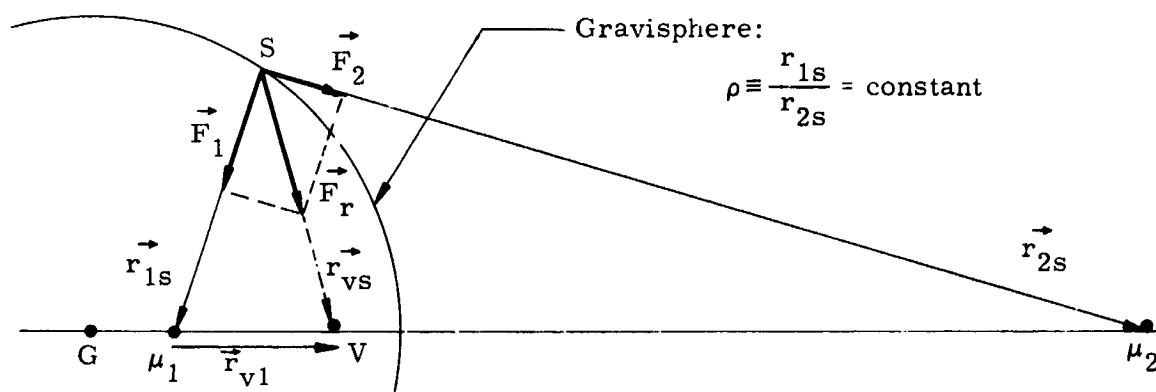


Fig. 2. The Gravispheric Force Center

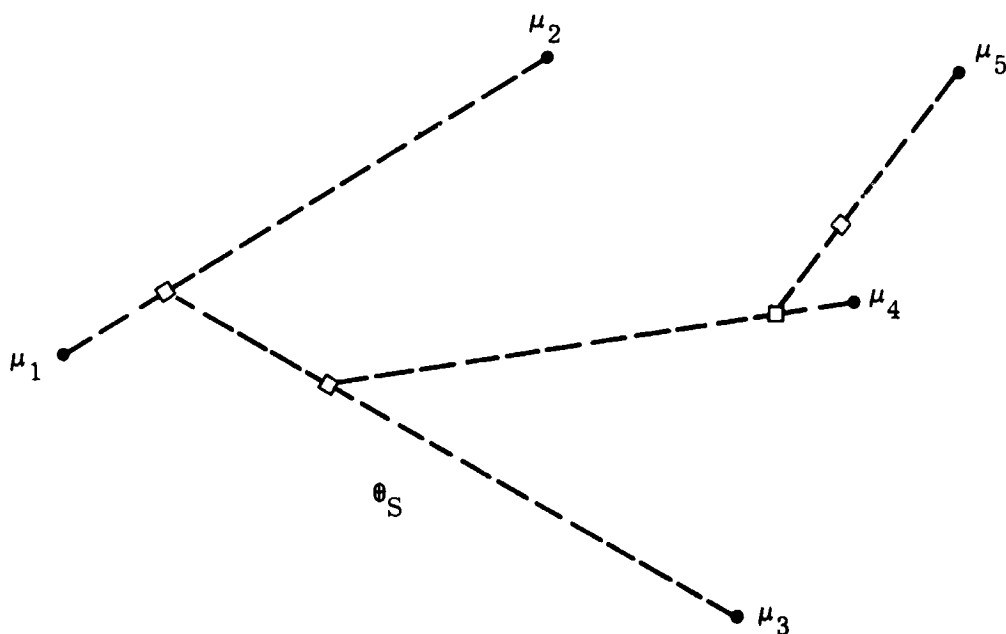


Fig. 3. Extension of Gravispheric Force Center Concept to More than Two Bodies

These considerations show how the attractions of two masses on an infinitesimal spacecraft can be reduced to the instantaneously equivalent attraction of a single mass. The magnitude and location of this equivalent mass on the line between the gravitating masses can be easily computed from equations (II-1, 2), knowing \vec{r}_s , \vec{r}_1 , \vec{r}_2 , μ_1 and μ_2 . Observe that when the spacecraft is equidistant from both bodies $r_{1s} = r_{2s}$ and Eq (II-1) reduces to the usual expression for the center of mass. Thus, only in this case does the gravispheric force center coincide with the barycenter. (In this case the gravisphere is the plane dividing the space between μ_1 and μ_2 .) Note also that the mass magnitude equals the total of the two real masses when the spacecraft is infinitely far displaced.

B. GENERALIZATION OF THE GRAVISPHERIC FORCE CENTER CONCEPT TO THE CASE OF MORE THAN TWO GRAVITATING BODIES

Extension of the concept of the gravisphere itself to the case of three or more bodies is impossible. Except under very special circumstances, there simply are no surfaces of constant ratios of distances or gravitational attractions. However, now that the expressions (II-1) and (II-2) have been derived, it is no longer necessary to think in terms of these surfaces used in the derivation. The simpler condition expressed by these relations suggests the method by which the concept can be extended to n bodies. Consider the geometry sketched in Fig. 3. First select any two masses (say μ_1 and μ_2) and via Eqs (II-1 and 2) replace them by an equivalent mass appropriate to the spacecraft position relative to them. Now take this fictitious mass and another one of the real gravitating bodies (μ_3 , say) and replace these two by a new fictitious mass. Continue this process, stepping around the system, until all gravitating masses have been replaced by a single equivalent mass.

This geometric description can be expressed analytically by a straightforward application of the formulas (II-1, 2). The first step, of course, yields

$$\vec{r}_{v12} = \frac{\frac{\mu_1 \vec{r}_1}{r_{1s}^3} + \frac{\mu_2 \vec{r}_2}{r_{2s}^3}}{\frac{\mu_1}{r_{1s}^3} + \frac{\mu_2}{r_{2s}^3}}$$

$$\mu_{v_{12}} = r_{v_{12}}^3 s \left(\frac{\mu_1}{r_{1s}^3} + \frac{\mu_2}{r_{2s}^3} \right)$$

where the subscripts 12 indicate that these values obtain for masses μ_1, μ_2 . Now again apply the basic formulas, treating $\mu_{v_{12}}$ as μ_1 , $\vec{r}_{v_{12}}$ as \vec{r}_1 and μ_3 as μ_2 , \vec{r}_3 as \vec{r}_2 :

$$\begin{aligned} \vec{r}_{v_{123}} &= \frac{\frac{\mu_{v_{12}} \vec{r}_{v_{12}}}{r_{v_{12}}^3} + \frac{\mu_3 \vec{r}_3}{r_{3s}^3}}{\frac{\mu_{v_{12}}}{r_{v_{12}}^3} + \frac{\mu_3}{r_{3s}^3}} \\ &= \frac{\frac{\mu_1 \vec{r}_1}{r_{1s}^3} + \frac{\mu_2 \vec{r}_2}{r_{2s}^3} + \frac{\mu_3 \vec{r}_3}{r_{3s}^3}}{\frac{\mu_1}{r_{1s}^3} + \frac{\mu_2}{r_{2s}^3} + \frac{\mu_3}{r_{3s}^3}} \end{aligned}$$

$$\begin{aligned} \mu_{v_{123}} &= r_{v_{123}}^3 s \left(\frac{\mu_{v_{12}}}{r_{v_{12}}^3} + \frac{\mu_3}{r_{3s}^3} \right) \\ &= r_{v_{123}}^3 s \left(\frac{\mu_1}{r_{1s}^3} + \frac{\mu_2}{r_{2s}^3} + \frac{\mu_3}{r_{3s}^3} \right) \end{aligned}$$

With repeated application of the procedure, one gets for n gravitating bodies:

$$\left.
\begin{aligned}
\vec{r}_v &= \frac{\vec{M}}{M_s} \\
\mu_v &= r_{vs}^3 M_s \\
\text{where} \\
\vec{M} &= \sum_{i=1}^n \frac{\mu_i \vec{r}_i}{r_{is}^3} \\
M_s &= \sum_{i=1}^n \frac{\mu_i}{r_{is}^3}
\end{aligned}
\right\} \quad (\text{II-3})$$

and where

μ_i = mass of ith gravitating body (times Universal Gravitation Constant)

\vec{r}_i = position of ith gravitating body

\vec{r}_s = position of spacecraft

$r_{is} = |\vec{r}_i - \vec{r}_s|$

$r_{vs} = |\vec{r}_v - \vec{r}_s|$

Equations (II-3) are very simple in form and represent the generalization of the gravispheric force center for two gravitating bodies to the case of n attractive masses. Since the concept of the gravisphere itself is inappropriate for the larger number of bodies, this generalized effective force center is called the "virtual mass."

Interchanging the indices in Eqs (II-3) does not alter the numerical values of these expressions. This independence of the order in which the physical masses are taken demonstrates the uniqueness of the virtual mass.

It is a simple matter to show that these equations for the virtual mass define a fictitious body which has the same effect upon the spacecraft as the combined effects of all the real bodies. Consider the

vector differential equation of motion of the spacecraft:

$$\ddot{\vec{r}}_s = \sum_{i=1}^n \frac{\mu_i (\vec{r}_i - \vec{r}_s)}{r_{is}^3}$$

This equation can be written as

$$\begin{aligned} \ddot{\vec{r}}_s &= \sum_{i=1}^n \frac{\mu_i \vec{r}_i}{r_{is}^3} - \vec{r}_s \sum_{i=1}^n \frac{\mu_i}{r_{is}^3} \\ &= \vec{M} - \vec{r}_s M_s \end{aligned}$$

by Eq (II-3c, d). By Eq (II-3a, b) it becomes

$$\ddot{\vec{r}}_s = M_s (\vec{r}_v - \vec{r}_s) = \frac{\mu_v}{r_{vs}^3} \vec{r}_{vs}$$

Thus, the virtual mass acceleration of the spacecraft is identical with the acceleration by the real gravitating bodies.

Equations (II-3) can be differentiated to give the velocity and mass rate of the virtual mass as functions of the positions and velocities of the spacecraft and the gravitating bodies:

$$\dot{\vec{M}} = \sum_{i=1}^n \frac{\mu_i}{r_{is}^3} \left[\dot{\vec{r}}_i - \vec{r}_i \left(\frac{V_{is}}{r_{is}} \right) \right] \quad (II-4)$$

$$\begin{aligned}
\dot{M}_S &= - \sum_{i=1}^n \frac{\mu_i}{r_{is}^3} \left(\frac{V_{is}}{r_{is}} \right) \\
\dot{\vec{r}}_V &= \frac{\dot{\vec{M}} - \vec{r}_V \dot{M}_S}{M_S} \\
\dot{\mu}_V &= \mu_V \left[\frac{V_{VS}}{r_{VS}} + \frac{\dot{M}_S}{M_S} \right]
\end{aligned}
\tag{II-4}$$

where

$$\frac{V_{is}}{r_{is}} = \frac{3}{2} \frac{\vec{r}_{is} \cdot \dot{\vec{r}}_{is}}{r_{is}}$$

C. THE SOLUTION TO THE N-BODY PROBLEM AS VIEWED IN THE LIGHT OF THE VIRTUAL MASS

It was shown in the preceding section that at any instant the virtual mass replaces the aggregate effect on the spacecraft of all the real gravitating bodies and thereby reduces the n-body problem to an unusual type of restricted two-body problem. This reduced problem is unusual in that the gravitating body does not remain in uniform motion but accelerates in inertial space and the mass magnitude varies. As Eqs (II-3) clearly show, whenever the spacecraft is very near to one of the real bodies (e. g., the jth one), that body's contribution to the virtual mass position and magnitude is highly favored (because of the division by the small r_{js}^3). In such a situation, the virtual mass is near to the dominant physical body ($\vec{r}_V \approx \vec{r}_j$) and essentially matches it in size ($\mu_V \approx \mu_j$). Slight differences occur due to the perturbing influences of the other bodies. As the trajectory carries the spacecraft far away from this real body and under the dominant influence of another one, the virtual mass continuously moves to the vicinity of the new body and grows or shrinks to nearly its mass magnitude. Thus, every spacecraft trajectory in an n-body gravity field has associated with it a separate phantom trajectory of the related virtual mass.

A simple example of this behavior is illustrated in Fig. 4. The trajectories shown are for the restricted three-body problem, where the two-dimensional circumlunar spacecraft trajectory is flown in the earth-moon orbital plane. Of course, for this case of only two gravitating bodies, the virtual mass motion is restricted along the earth-moon line. The two paths are depicted as the solid lines in an inertially oriented barycentric coordinate system. The moon trajectory is shown, however, the earth motion has been omitted to keep the curves uncluttered near the origin. Relative position lines between the virtual mass and the spacecraft are shown at several time points by the dashed lines. To the scale of the plot, the initial virtual mass displacement from the center of earth is indistinguishable. Note also that the virtual mass coincides with the barycenter at approximately 22 hr, where the spacecraft is equidistant from earth and moon. Figure 5 shows the corresponding variation of the virtual mass magnitude for this example. The abscissa is the virtual mass displacement along the earth-moon line. Time points corresponding to those appearing in Fig. 4 are spotted on the curve.

Of course, the idea is immediately suggested of using the virtual mass as a means of constructing the spacecraft n-body trajectory in a stepwise numerical procedure. Consider that the spacecraft position and velocity are given in some reference frame at some instant of time. Assume also that an ephemeris gives the positions and velocities of the gravitating bodies (of known masses) in this same reference frame. These data are sufficient to compute the virtual mass position, velocity, mass magnitude and magnitude rates from Eqs (II-3) and (II-4). Then by simple subtractions, the spacecraft position and velocity vectors can be computed relative to the virtual mass at this instant of time. If now the relative motion is computed over some increment of time, the spacecraft trajectory can be propagated and transformed back to the reference coordinate frame. The whole process can now be repeated with the new position and velocity of the vehicle at the new time.

If the virtual mass were fixed in magnitude and unaccelerated, one could compute the spacecraft relative motion over any finite arc with no error as the conic section solution to the two-body problem. The absolute motion would be exact as well for this case where the fixed magnitude virtual mass moves with constant velocity. The mass and velocity do change, however, and hence, the characterizations of the spacecraft relative motion as a conic section and of the virtual mass magnitude and velocity as constant are not exact. But this is no different from any other approximation scheme associated with the numerical integration of differential equations. The fundamental theorem of the calculus guarantees that theoretically, the errors of this approximation will vanish in the limit as the arc length (time increment) approaches zero. There is, of course, a practical limit to the accuracy which can be achieved due to the limitation of the number

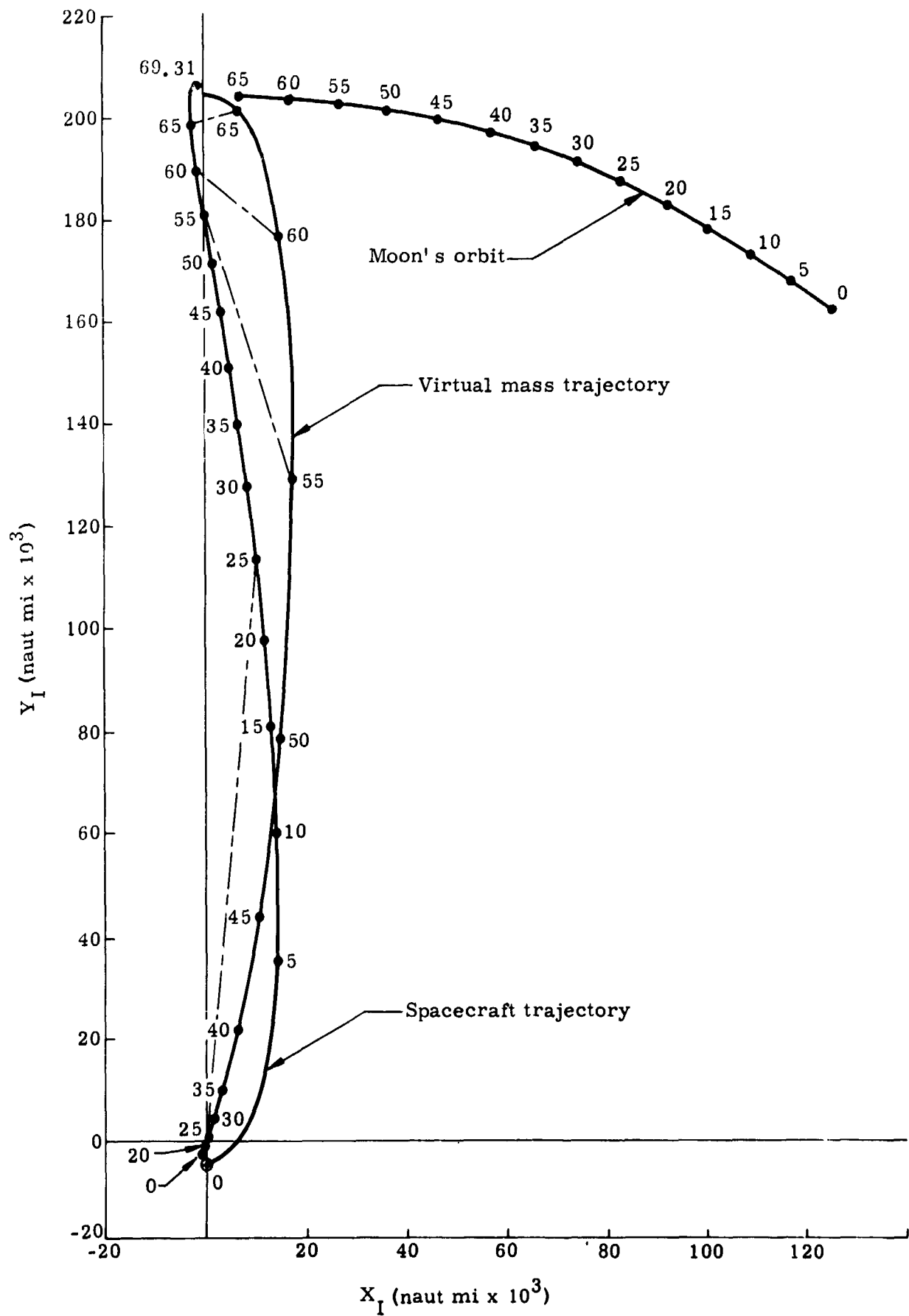


Fig. 4. Virtual Mass Trajectory for an In-Plane Spacecraft Trajectory

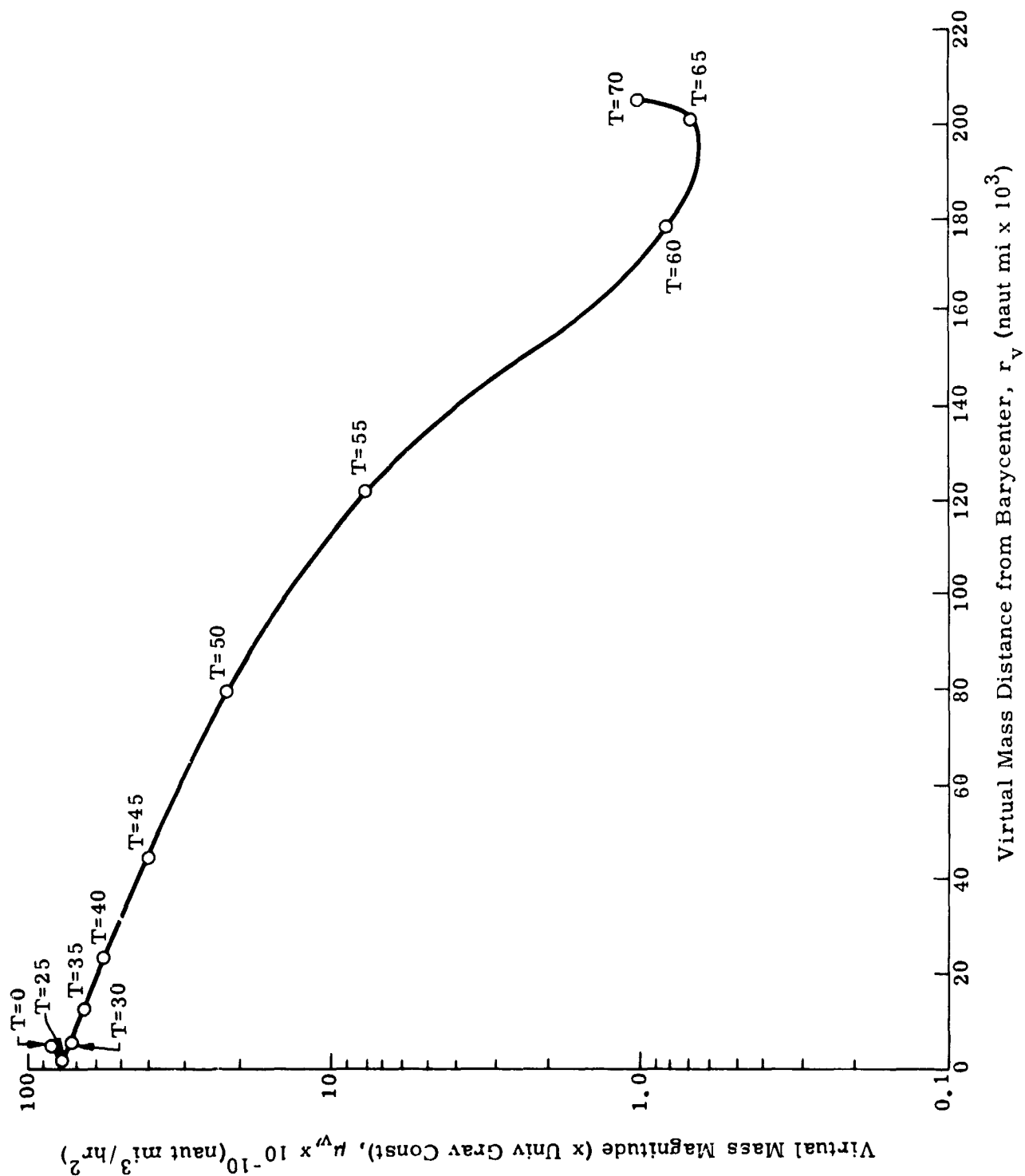


Fig. 5. Virtual Mass Magnitude Versus Distance from the Barycenter

of digits which can be carried in the computations and the length of time available to perform them. The next two chapters will treat these practical aspects of the numerical calculation.

This chapter will be concluded with some observations concerning this new procedure for solving the n-body problem. There is a similarity to the Cowell method in the procedure of adding up the attractions of all the real gravitating bodies at each computing step (see Eqs (II-3)). This summation, however, is not expressed in terms of the resultant force; but rather as the magnitude and location of a "virtual mass" which instantaneously produces identically the same resultant force on the spacecraft. It is like the Encke procedure in that a Keplerian conic section is computed relative to the virtual mass as the reference body. Of course, there are no discontinuous jumps from one reference body to another, since the virtual mass moves continuously from the vicinity of one real body to that of another as the spacecraft trajectory is dominated by successively different bodies. Since all the perturbing effects are included in the computation of the virtual mass, a perfect rectification is made at each computing interval. This then eliminates entirely the need for numerically integrating higher order acceleration perturbations. Thus, finally, the procedure is like the patched conic technique in that only preintegrated conic section solutions are pieced together.

III. DIGITAL COMPUTATION

FORMULATIONAL CONSIDERATIONS

It has been truly said that numerical computation is more of an art than a science. This Chapter in fact is an exposition of a primitive form of the art practiced here to implement the concepts discussed in the last Chapter in a digital computer program. Where alternative approaches and variable mechanizations are described here, they were tested and compared in the computer. The results are reported in the following Chapter.

A. VECTOR ORBITAL ELEMENTS

A number of complications and inefficiencies would result if the computation scheme outlined in the preceding Chapter were implemented in terms of the conic section equations as generally written in polar coordinates in the plane of motion. The complications would arise in the special procedures required to handle cases of zero inclination, zero eccentricity and unity eccentricity. The principal inefficiency would manifest itself in the necessity for a large number of coordinate transformations. Each computation cycle would require a rotational transformation from the reference (ephemeris) frame to the instantaneous plane of motion, defined by the position and velocity relative to the virtual mass, and back again.

The transformations can be eliminated entirely and the other difficulties minimized by using the three-dimensional vector formulation of the two-body conic section solution. These relations will be developed here for the sake of including in this report a complete listing of the equations required for the computation.

If both sides of the vector equation of motion for the two-body problem: *

$$\ddot{\vec{r}} = -\frac{\mu}{r^3} \vec{r} \quad (\text{III-1})$$

are cross-multiplied by \vec{r} , the equation

$$\vec{r} \times \ddot{\vec{r}} = -\frac{\mu}{r^3} \vec{r} \times \vec{r} = 0$$

results.

*The quantities are not subscripted here for the sake of simplicity of notation. It is to be understood, nevertheless, that the spacecraft motion relative to the virtual mass is implied.

This can be integrated to obtain

$$\vec{k} = \vec{r} \times \dot{\vec{r}} \quad (\text{III-2})$$

The constant of integration \vec{k} will be called the "kepler vector" since it obviously represents twice the areal rate. Now form the vector product of Eq (III-2) and Eq (III-1), divided by $-\mu$:

$$-\frac{1}{\mu} \vec{k} \times \ddot{\vec{r}} = \frac{(\vec{r} \times \dot{\vec{r}}) \times \ddot{\vec{r}}}{r^3}$$

It can easily be shown that the right side is $\frac{d}{dt} \left(\frac{\vec{r}}{r} \right)$ and hence this equation can be integrated to yield

$$\vec{e} = -\frac{\vec{r}}{r} - \frac{\vec{k} \times \dot{\vec{r}}}{\mu} \quad (\text{III-3})$$

This integration constant \vec{e} will be called the "eccentricity vector." The magnitude of \vec{e} is the eccentricity of the conic section and the vector points along the major axis toward periapsis.

The equation of the conic section is easily derived from Eq (III-3) by forming its inner product with \vec{r} :

$$\vec{e} \cdot \vec{r} = -\frac{\vec{r}}{r} \cdot \vec{r} - \frac{\vec{k} \times \dot{\vec{r}}}{\mu} \cdot \vec{r}$$

Interchanging the dot and cross in the last term on the right and substituting from Eq (III-2) gives finally

$$\vec{e} \cdot \vec{r} = -r + \frac{k^2}{\mu} \quad (\text{III-4})$$

Actually Eq (III-4) defines a three-dimensional surface rather than a path. The orbit is specified as the intersection of this surface with the plane normal to \vec{k} .

The velocity $\dot{\vec{r}}$ can easily be determined at any position \vec{r} on a given orbit \vec{k} , \vec{e} . Observe first that since \vec{k} is orthogonal to $\dot{\vec{r}}$:

$$\frac{\vec{k}}{k} \times \dot{\vec{r}}$$

is a vector in the plane of motion, perpendicular to the velocity vector and equal to it in magnitude. The cross product of this resulting vector

by the same unit normal to the plane gives the original velocity identically:

$$\dot{\vec{r}} \equiv \left(\frac{\vec{k} \times \dot{\vec{r}}}{k} \right) \times \frac{\vec{k}}{k} = \frac{\vec{k}}{k^2} \times \left(-\vec{k} \times \dot{\vec{r}} \right)$$

Substitute for the expression in parentheses from Eq (III-3) to obtain

$$\dot{\vec{r}} = \frac{\vec{k}}{k^2} \times \mu \left(\vec{e} + \frac{\vec{r}}{r} \right) = \left(\frac{\mu}{k^2} \right) \vec{k} \times \left(\vec{e} + \frac{\vec{r}}{r} \right) \quad (\text{III-5})$$

\vec{k} and \vec{e} are completely determined in any three-dimensional coordinate system by Eqs (III-2) and (III-3), having given the position \vec{r} , velocity $\dot{\vec{r}}$ and central mass μ . These vectors define the geometry of the orbit just as do the classical orbital elements a, e, i, Ω, ω . Of course, six elements are defined by the three components each of \vec{k} and \vec{e} , but the identical satisfaction of the orthogonality condition

$$\vec{e} \cdot \vec{k} \equiv 0$$

implies that, in fact, there are only five independent elements.

The behavior of the \vec{k} and \vec{e} orbital elements of the spacecraft motion relative to the virtual mass is illustrated in Figs. 6 and 7 for the example circumlunar trajectory of Chapter II, Section C. Recall that, in this simple case, the motion is two-dimensional in the earth-moon orbital plane. Therefore, the \vec{k} vector is everywhere orthogonal to this plane and hence its magnitude variation (shown in Fig. 7) is the only significant feature. The eccentricity vector, on the other hand, lies in the plane and varies in both magnitude and direction. Figure 6 depicts \vec{e} as a series of arrows, emanating from the virtual mass focal points, pointing in the indicated directions and equal in lengths to the eccentricities appropriate to the positions.

This section is concluded with an explanation of the direct method for computing the conic section time of flight from given initial position \vec{r}_1 to final position \vec{r}_2 on a known orbit:

$$t_2 - t_1 = f(\vec{r}_1, \vec{r}_2, \vec{k}, \vec{e})$$

No derivations are given. Known results are simply expressed in terms of the vector notation adopted here.

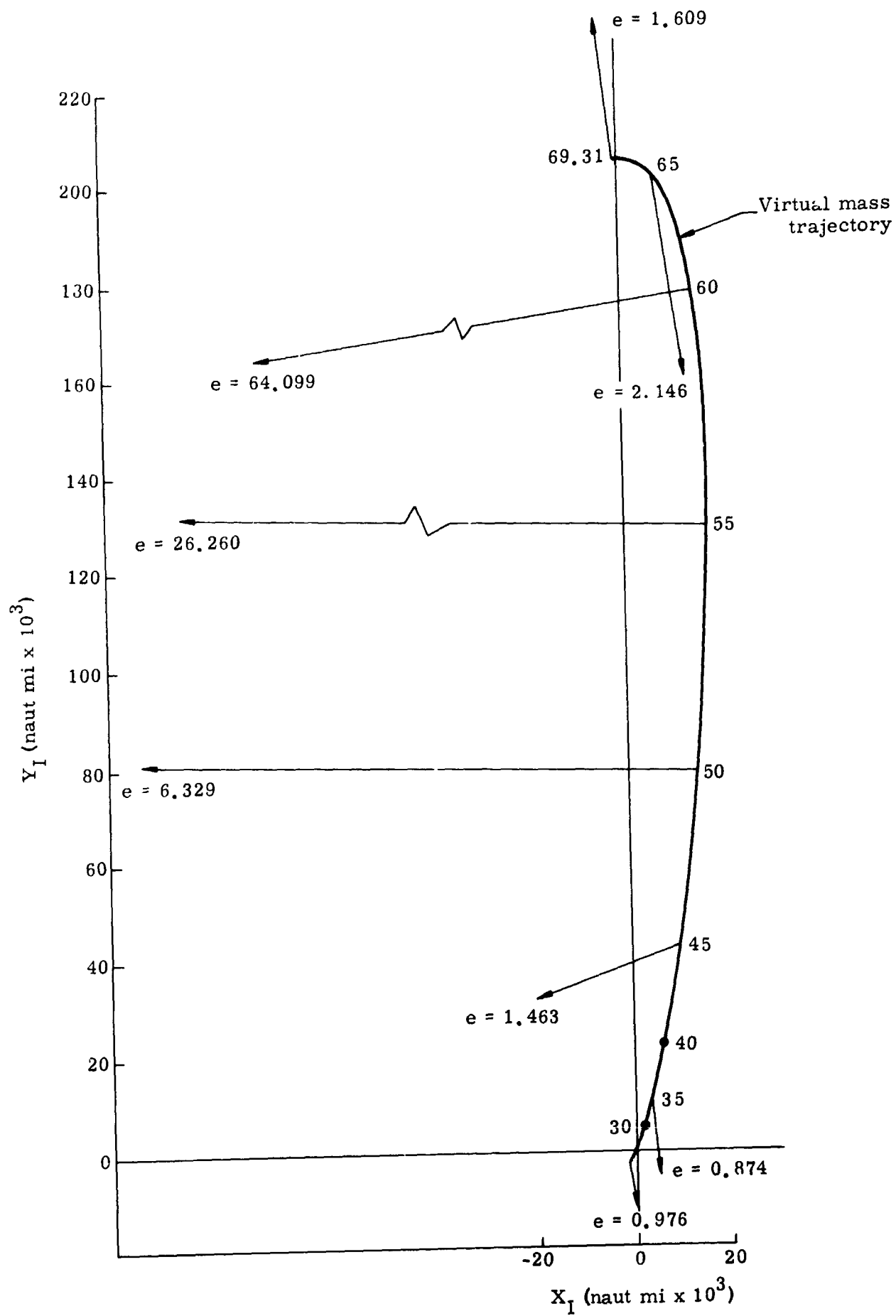


Fig. 6. Eccentricity Vector for Various Virtual Mass Positions for In-Plane Trajectory

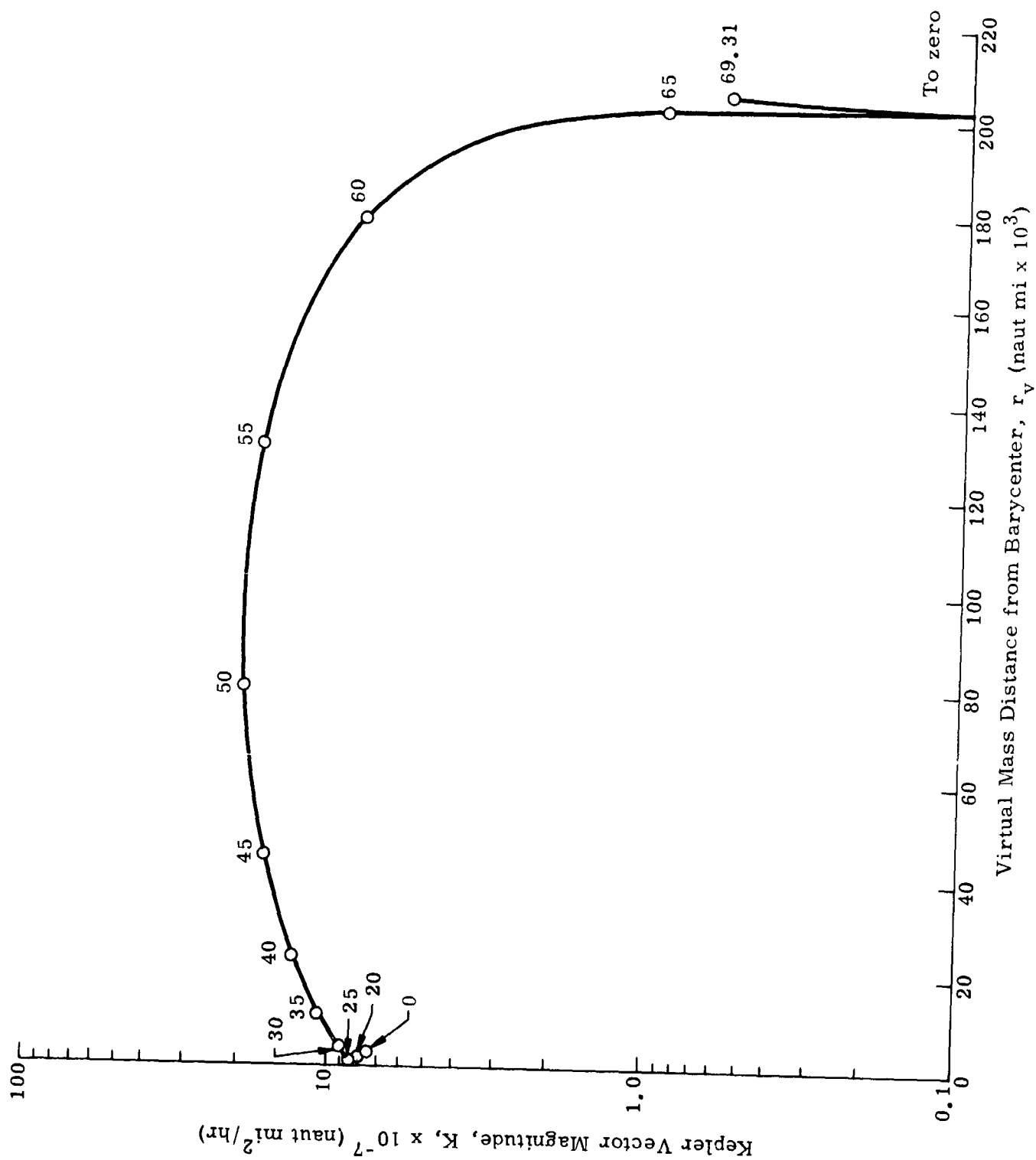


Fig. 7. Kepler Vector Magnitude Versus Virtual Mass Displacement from Barycenter for In-Plane Trajectory

Section C of this Chapter describes how to handle the inverse problem of finding the final position \vec{r}_2 on a given orbit, with a prescribed flight time from an initial position \vec{r}_1 :

$$\vec{r}_2 = \vec{g}(\vec{r}_1, \vec{k}_1, \vec{e}, t_2 - t_1)$$

The conic section time of flight can be computed from

$$t_2 - t_1 = \frac{M_2 - M_1}{\omega_M} \quad (\text{III-6})$$

In this expression, when the orbit is elliptic or hyperbolic ($e \neq 1$), M is interpreted as the mean anomaly and ω_M as the mean angular rate. For the parabolic case ($e = 1$), M is taken to be the area swept out by the radius vector as it rotates from periapsis and ω_M the (constant) areal rate. The value M can be represented in the algebraic form

$$M_i = E_i - \psi_i \quad (i = 1, 2) \quad (\text{III-7})$$

in all cases. When $e \neq 1$, E represents the eccentric anomaly and $\psi = e \sin E$ or $e \sinh E$. In the hyperbolic case the sign of M should be reversed; but, as will be shown later, this can be accommodated in the sign of ω_M . When $e = 1$, E represents the area obtained by projection of the parabolic arc normal to the major axis and ψ defines the triangular area obtained by similar projection of the radius vector to the position defining the end of the arc. The parabolic triangular area is signed negatively when the true anomaly is less than 90° so that Eq (III-7) is always valid.

It remains now to show how the values of E , ψ , and ω_M are computed for the various cases.

First, some preliminary computations are defined. The in-plane unit normal to the major axis is

$$\left. \begin{aligned} \vec{n} &= \frac{\vec{k} \times \vec{e}}{k e} & (\vec{e} \neq 0) \\ \vec{n} &= \frac{\vec{k} \times \vec{r}_1}{k r_1} & (\vec{e} = 0) \end{aligned} \right\} \quad (\text{III-8})$$

Note that in the circular case the major axis is arbitrarily assumed along the initial position vector. The length of the semi-minor axis is

$$\left. \begin{aligned} b &= \frac{k^2}{\mu (|1 - e^2|)^{1/2}} & (e \neq 1) \\ b_i &= \frac{2}{r_i - k^2/\mu} & (e = 1) \end{aligned} \right\} \quad (\text{III-9})$$

The semi-minor axis is infinite in the parabolic case, hence Eq (III-9b) is written to give the reciprocal of one-half the base of the aforementioned triangular area (the denominator is $-\vec{e} \cdot \vec{r}$ by Eq (III-4)). The projection of the radius vector orthogonal to the major axis, divided by b , is simply

$$X_i = \frac{\vec{n} \cdot \vec{r}_i}{b_i} \quad (\text{III-10})$$

These auxiliary computations now make it easy to display the necessary values. First ω_M is given by

$$\left. \begin{aligned} \omega_M &= \frac{\mu (1 - e^2)}{kb} & (e \neq 1) \\ \omega_M &= \frac{k}{2} & (e = 1) \end{aligned} \right\} \quad (\text{III-11})$$

The first value represents the mean angular rate, the second is the areal rate. As noted earlier, $\omega_M < 0$ for hyperbolic orbits ($e > 1$). The value of ψ is given by

$$\psi_i = e X_i \quad (\text{III-12})$$

in all cases. Note that when $e = 0$, $\psi_i = 0$ (or $M_i = E_i$) and that ψ_i indeed is the triangular area for $e = 1$ (by Eqs (III-9b) and (III-10)). Finally, the eccentric anomaly (or parabolic arc area) E is

$$\left. \begin{aligned} E_i &= \sin^{-1} X_i & (e < 1) \\ E_i &= \frac{(k^2/\mu \cdot X_i)^3}{3} & (e = 1) \\ E_i &= \sinh^{-1} X_i & (e > 1) \end{aligned} \right\} \quad (\text{III-13})$$

There is no ambiguity in the hyperbolic case since the orbit is aperiodic. This is reflected in the fact that the inverse hyperbolic sine is a monotonically increasing function of the argument. The ambiguity which does exist in the periodic elliptic case can be easily resolved. When $e \neq 0$

$$\left. \begin{aligned} E_i &= \text{principal value} = PV \quad \text{for } r_i \leq a \\ E_i &= \pi - PV \quad \text{for } X_i > 0, r_i > a \\ E_i &= -\pi - PV \quad \text{for } X_i < 0, r_i > a \end{aligned} \right\} \quad (\text{III-14})$$

When $e = 0$, the above test on $r - a$ must be replaced by a test on $\vec{r}_1 \cdot \vec{r}_2$.

Note that the time can be negative in the case where $e < 1$ and the cut $E = \pi$ (or $-\pi$) is crossed. If this should happen merely add $2\pi/\omega_M$ to the time given by Eq (III-6).

B. NONITERATED VERSUS ITERATED COMPUTATION

The characterization of the virtual mass motion as a constant-velocity straight line and of the mass magnitude as held constant over each computing interval is dynamically consistent with the characterization of the spacecraft relative motion as a conic section. Therefore, an important problem concerning the computation is the determination of a method for establishing appropriate values of the virtual mass velocity and mass to hold constant over the interval.

The simplest approach, of course, is to merely take the values given by the virtual mass equations themselves at the beginning of the step. These values can be used, much as in the classical Euler integration scheme, to propagate the motion to the end of the interval, where new values are discontinuously assumed consistent with the new situation. This procedure is fast since just one computation (no iteration) is required per time interval. Unfortunately, accuracy suffers due to the fact that initial values, rather than mean values, are used over the step. Whereas the spacecraft trajectory itself would be continuous in this case, perhaps the most serious failing would result from the discontinuities in the virtual mass trajectory. The virtual mass position propagated to the end of an interval, for the purpose of locating the spacecraft, would not, in general, correspond to the position computed by Eq (II-3) for the start of the next interval.

If the correct position and magnitude of the virtual mass at the end of the interval were known a priori, there would be no problem whatever in establishing the required average velocity or in choosing some linearly interpolated value of the mass to hold constant:

$$\dot{\vec{r}}_{v_{av}} = \frac{\vec{r}_{v_e} - \vec{r}_{v_B}}{\Delta t}$$

$$\mu_{v_{av}} = C_1 \mu_{v_e} + (1 - C_1) \mu_{v_B} \quad (0 \leq C_1 \leq 1 \text{ is a specified constant})$$

(III-15)

Since these final values are not known at the outset, but are in fact part of the answer sought, an iterative computation procedure, analogous to the modified Euler scheme, is suggested. The final values \vec{r}_{v_e} and

μ_{v_e} are estimated initially and then iteratively improved by computation

based upon the resulting spacecraft final position. When the difference between successive values becomes acceptably small, the iteration can be discontinued and the computation can proceed to the next interval. The better the initial estimate, naturally, the faster the convergence. The method decided upon for study was to assume a second order variation with time in computing this first guess.

$$\left. \begin{aligned} \vec{r}_{v_e} &= \vec{r}_{v_B} + \dot{\vec{r}}_{v_B} (\Delta t) + \ddot{\vec{r}}_{v_{av}} (\Delta t)^2 \\ \mu_{v_e} &= \mu_{v_B} + \dot{\mu}_{v_B} (\Delta t) + \ddot{\mu}_{v_{av}} (\Delta t)^2 \end{aligned} \right\} \quad \text{(III-16)}$$

The constant terms are given by Eqs (II-3), the linear term coefficients by Eqs (II-4). The (acceleration) coefficients of the squared terms are assumed to hold for this interval from the previous one. Thus, they would be computed as

$$\left. \begin{aligned} \ddot{\vec{r}}_{v_{av}} &= \frac{\vec{r}_{v_e} - \vec{r}_{v_B} - \dot{\vec{r}}_{v_B} (\Delta t)}{(\Delta t)^2} \\ \ddot{\mu}_{v_{av}} &= \frac{\mu_{v_e} - \mu_{v_B} - \dot{\mu}_{v_B} (\Delta t)}{(\Delta t)^2} \end{aligned} \right\} \quad \text{(III-17)}$$

after convergence was achieved in the preceding interval. At the starting time step, they are set to 0. Although this iterative scheme is slightly more complicated and requires multiple looping each interval, there will be no discontinuity in the virtual mass trajectory, and accuracy should be better, for a given step size, than with the simple non-iterative approach.

C. THE COMPUTING INTERVAL

It is intuitively obvious that different computing interval sizes are required for different parts of a trajectory. Large step sizes can be used when the spacecraft is far away from a relatively constant magnitude and slowly moving virtual mass (such as would be the case during the heliocentric arc of an interplanetary mission). Small increments should be taken, however, whenever the vehicle is close to the virtual mass (as for a trajectory grazing by a planet) or whenever a sphere-of-influence crossing occurs and the virtual mass moves and changes magnitude rapidly from one dominant physical body to another.

If the step size is controlled to maintain equal increments of true anomaly in the motion relative to the virtual mass, the time increment variation will behave qualitatively as desired. The simple formula for converting the true anomaly increment into the corresponding time interval is

$$\Delta t = \frac{C_2 r_{vs}^2}{k} \quad (\text{III-18})$$

where k is the magnitude of \vec{k} and where C_2 is an input constant defining the desired angular step size in radians. Multiplication of C_2 by r_{vs}^2 , of course, converts the angle into twice the area increment. Dividing by double the instantaneous areal rate relative to the virtual mass gives (for small steps) the time to cover this angle.

A practical difficulty could arise in attempting to use Eq (III-18) as it is. Figure 7 shows that k vanishes at one point along the trajectory for the in-plane case. At this point, of course, the relative motion is directly toward or away from the virtual mass. Although k does not vanish for more general trajectories, it still can become small enough to cause Eq (III-18) to compute a very large time increment. This problem can be circumvented by replacing k by $r_{vs} V_{vs}$, the scalar product of the position and velocity magnitudes. This then represents a fictitious areal rate which assumes that the velocity is always normal to the position vector and thus, in general, is larger than the true areal rate. Substituting this into Eq (III-18) gives

$$\Delta t = \frac{C_2 r_{vs}}{V_{vs}} \quad (\text{III-19})$$

as the time increment. This form can give computational difficulty only when $V_{vs} \rightarrow 0$ --a highly unlikely occurrence.

It is one thing to set the desired time increment, but the realization of it is another matter. Since the conic section time of flight is a transcendental form, it is not possible to invert it to determine in closed form the final position corresponding to a given flight time from a given initial position. Two alternatives are possible, depending upon whether the basic computation philosophy is noniterative or iterative (see Section B). In the noniterative approach, the final position is estimated so as to approximate the desired time. Once the estimate is made, the desired time is disregarded and the conic section time of flight equations are used to ascertain what time actually did elapse from \vec{r}_1 to the estimated \vec{r}_2 . The trajectory time and ephemeris time are then updated by this true time increment in preparation for the next step.

The iterative procedure cannot be treated so simply, however, because the procedures for estimating and updating the final values of the virtual mass and for computing the average velocity all depend upon achieving a predetermined time increment with very high accuracy. A double iteration could be mechanized in which the spacecraft final position is iterated within the outer loop of the virtual mass final condition iteration. Such a procedure is cumbersome and time-consuming. It is also unnecessary if accurate initial estimates of both the spacecraft and the virtual mass final conditions can be made and then simultaneously updated within a single iteration loop. This latter course was decided upon for the mechanization of the iterated virtual mass procedure. The logical details of the technique are not of primary concern here and, hence, are deferred until Chapters V and VI. The establishment of the computation interval is the subject of interest here.

It has been shown that an accurate estimation procedure for the spacecraft final relative position is required for both the noniterative and the iterative approaches. Since this estimate, itself, will be repeatedly applied in the iterative scheme, the objective is to develop an estimation procedure which improves with each iteration.

The spacecraft and virtual mass data are known at the beginning of the interval. The virtual mass magnitude and velocity are given by Eqs (II-3) and (II-4) for the noniterated case or by Eqs (III-15) for the iterated case. The initial relative position and velocity and the mass, therefore,

are trivially determined and from them, the vector orbital elements \vec{k} , \vec{e} are obtained by Eqs (III-2) and (III-3). Equation (III-19) gives the desired step size, Δt . The final position \vec{r}_{vs_2} must lie in the plane of relative motion defined by \vec{r}_{vs_1} , $\dot{\vec{r}}_{vs_1}$ and, hence, can be expressed as a linear combination of them:

$$\vec{r}_{vs_2} = B \left[\vec{r}_{vs_1} + (\Delta\tau) \dot{\vec{r}}_{vs_1} \right] \equiv B \vec{\sigma}_{vs_2} \quad (\text{III-20})$$

The geometry is illustrated in Fig. 8 and shows that $\Delta\tau$ determines the time (or true anomaly) increment and B ensures satisfaction of the orbital equation. Once $\Delta\tau$ is given, B is easily computed since Eq (III-20) must satisfy Eq (III-4):

$$B = \frac{k^2/\mu_v}{\vec{e} \cdot \vec{\sigma}_{vs_2} + \sigma_{vs_2}} \quad (\text{III-21})$$

The question therefore is reduced to that of relating $\Delta\tau$ to the desired Δt . As in the case of the virtual mass estimation procedure, a second order variation will be assumed. Here the constant term is 0 and the linear coefficient is 1, for it must be true that $\Delta\tau \rightarrow \Delta t$ as $\Delta t \rightarrow 0$

$$\Delta\tau = \Delta t + \kappa (\Delta t)^2 \quad (\text{III-22})$$

The procedure for evaluating κ is similar to that used for the second order coefficients in Eq (III-16). After the computation of the conic section time of flight $\Delta t_k = t_2 - t_1$ (from Eq (III-6)) in the preceding iteration, that Δt_k value and the $\Delta\tau$ value used to obtain it specify the exact κ for that case as

$$\kappa = \frac{\Delta\tau - \Delta t_k}{(\Delta t_k)^2} \quad (\text{III-23})$$

This value will be assumed to hold for the present iteration from the last one. Clearly, this assumption gets better and better as $\Delta t_k \rightarrow \Delta t$, the desired time increment. In the noniterated case κ is merely updated to provide the best first (and only) estimate of the next interval.

The study reported in Ref. 2 showed that, under some circumstances, the conic section time of flight may be different from the true time. In the event such a time bias might prove desirable in this case, the provision was made to cause $\Delta t_k \rightarrow C_3 \Delta t$ by rewriting Eq (III-22) as

$$\Delta \tau = C_3 \Delta t + \kappa [C_3 \Delta t]^2 \quad (\text{III-24})$$

where C_3 is an input constant.

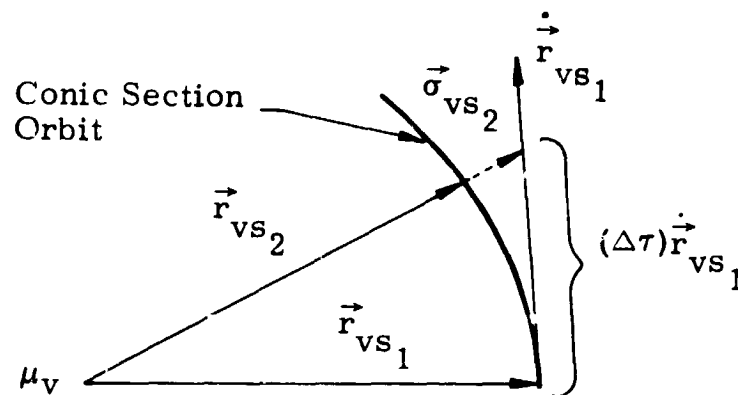


Fig. 8. Geometry of Spacecraft Final Relative Position Determination

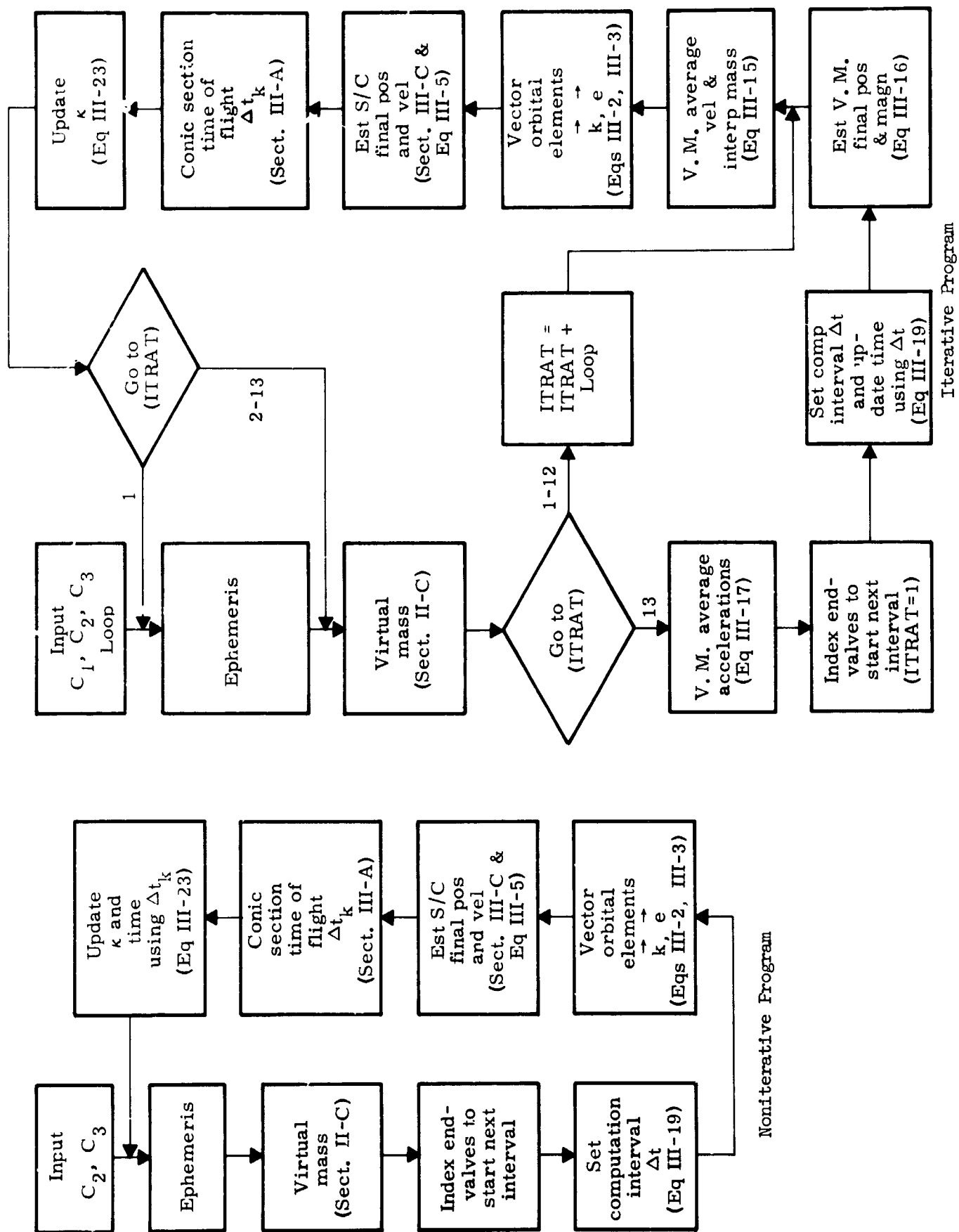


Fig. 9. Iterative and Noniterative Program Logic

IV. DIGITAL COMPUTER STUDY

A. APPROACH

The suitability of the virtual mass technique as a flexible integration method for the n-body problem could be assessed only by trying some numerical examples. Accordingly, the basic concept described in Chapter II was mechanized, in conformity with the considerations of Chapter III, as two separate computer programs: one a simple noniterative procedure, the other a somewhat more complicated iterative procedure. Salient features of the flow diagrams for the two programs are sketched in Fig. 9. Details such as special logic paths for starting the computation and tests for stopping conditions and printout have been omitted in order to emphasize the basic principles of operation. Reference is made on the flow diagrams to the sections of the two previous chapters where the appropriate equations may be found. Note that the iterated program loops through the ephemeris subroutine only once each computation interval. Since the desired time increment Δt is fixed and the iteration procedure is intended to make repetitive improvement to achieve this objective, the final time and, hence, the gravitating body data are fixed. Improvement in the virtual mass data, therefore, is effected by improvement in the spacecraft final position and velocity.

The input constants provide the means by which the computations are controlled within the programs. C_2 sets the desired computation interval size (see Section III-C). C_3 biases the Keplerian flight time for values different from unity (see Section III-C). In the iterated program C_1 ($0 \leq C_1 \leq 1$) linearly interpolates the virtual mass magnitude to some value between the initial and final values (see Section III-B). The constant LOOP controls the number of iterations per computing interval according to the following:

<u>Value of LOOP</u>	<u>No. of iterations (after first pass)</u>
12	1
6	2
4	3
3	4
2	6
1	12

In order to properly assess the effects of variations of the program controls and to compare the two programs with each other, an index to the accuracy of the solution is necessary. The constancy of the Jacobi integral is a necessary condition to any solution to the restricted three-body problem and, therefore, could be used for just such an accuracy index.

In addition, this case of just two gravitating bodies is the simplest n-body problem and would serve adequately as a test of the integration method. Therefore, the ephemeris subroutine was programmed for the restricted three-body problem by representing two bodies in circular orbits about their common center of mass. The expression for the Jacobi integral is classically derived in the rotating barycentric coordinate system. Since all computations for the virtual mass procedure are carried out in an inertially oriented barycentric frame, the Jacobi constant was transformed to that reference;

$$C_J = 2 \left(\frac{\mu_1}{r_{1s}} + \frac{\mu_2}{r_{2s}} \right) - (\dot{x}_s)^2 - (\dot{y}_s)^2 - (\dot{z}_s)^2 - 2\omega (y_s \dot{x}_s - x_s \dot{y}_s)$$

It was recognized that some of the computations represented by the equations in Chapters II and III may involve differences between nearly equal numbers. Loss of significance in such cases can be alleviated by carrying out these computations in double precision. Rather than attempt an analysis in detail to isolate those computations where increased accuracy would be required, all computations were done in double precision on the IBM 7094 digital computer.

A circumlunar trajectory, inclined initially nearly 30° to the earth-moon plane, was chosen as the principal test trajectory. The pericynthion altitude of about 210 naut mi (lunar radius ≈ 938.5 naut mi) was reached in slightly more than 70.3 hr from insertion at earth. This trajectory is given in Chapter V as a sample problem solved by the final version of the program. All the details, including the initial conditions, the physical constants describing the earth-moon system and the trajectory time-history, appear there.

B. RESULTS

Although a great number of exploratory studies and parametric runs had to be made, the pertinent results can be summarized quite concisely.

As expected, the iterated program was more accurate than the simple noniterated one. A direct comparison of the two is shown in Fig. 10 in terms of the Jacobi energy accuracy index. The gains controlling the computing interval size, $C_2 = 0.0005$, and the time bias, C_3 , were set to the same values for the two programs. The curve shows the difference between the Jacobi energies at corresponding time points on the test trajectory as computed by the two programs. This method of presentation was chosen because, although the gains selected caused the iterated program to compute with high accuracy, there was a small variation of the Jacobi energy. The difference shown in the curve of Fig. 10 shows how much worse the variation was using the noniterative

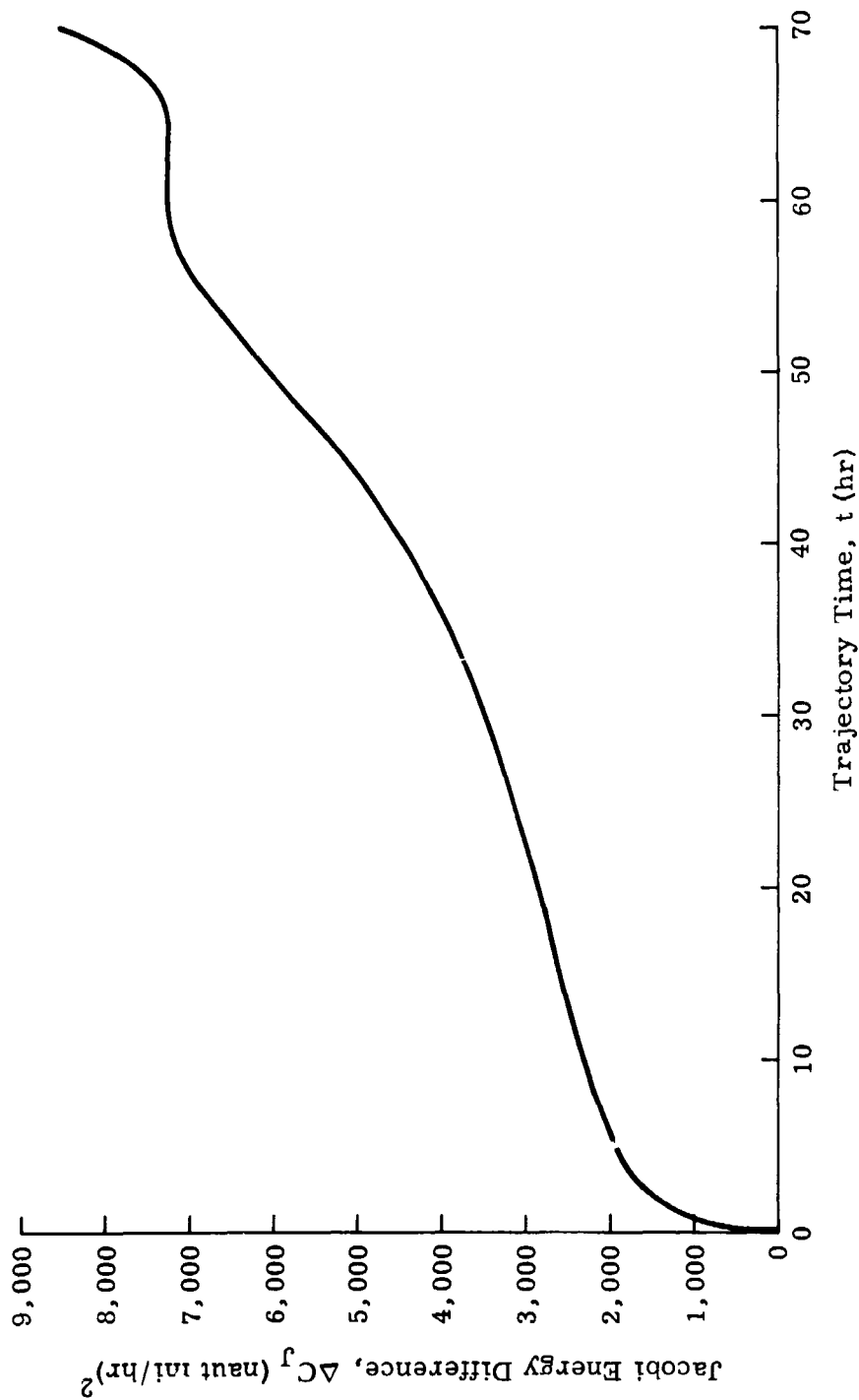


Fig. 10. Variation of Noniterated Trajectory Jacobi Energy from Iterated Solution for Same Gain C_2 on Computation Interval Size

program. The maximum difference was an appreciable $9000 \text{ (naut mi/hr)}^2$ out of $7033989.7388 \text{ (naut mi/hr)}^2$. As will become more apparent in the other studies of the iterative program, the superior performance of the more sophisticated method well justifies the slight additional complication. Without it, the calculation of precision trajectories would be impractical.

Restricting attention, therefore, to the iterative program, the first question to be resolved was that of the number of iterations required per computing interval. With any reasonable gain C_2 on the computing interval size, it was found that just one extra loop (after the first pass) was sufficient to produce answers which, to at least eight significant figures, were identical with those for more repetitions. Therefore, the final program has been fixed to loop the first time through the computations, including an access to the ephemeris subroutine, and then just one additional iteration (by-passing the ephemeris).

The studies of the interpolated virtual mass value and the conic section time bias are summarized in Figs. 11 and 12. For both of these comparisons, a base run was made with the mass interpolated at the midpoint ($C_1 = 0.5$) and no time bias ($C_3 = 1.0$). The curve of Fig. 11 was generated as the time-history of the differences in the Jacobi energy between the base run and two others in which $C_1 = 0.4$ and 0.6 . The incremental error buildup portrayed by this graph clearly shows that the best mass value to use is the arithmetic mean between the initial and final values. Similarly, the incremental error curve of Fig. 12 was generated as deviations from the base run of two time biases of $C_3 = 0.99999$ and 1.00001 . This curve shows the extreme sensitivity of the solution accuracy to this parameter and indicates that no bias ($C_3 = 1.0$) is best. On the basis of these results, the final version of the program is coded to calculate the virtual mass magnitude as the simple average of the end-values and to achieve an unbiased match between the desired and the conic section time increments.

The constant C_2 , since it controls the computation interval size, is the basic accuracy selector. Final comparative studies of this parameter were made with the other program controls set to the optimized values as described above. The variation of the Jacobi energy with time along the test trajectory is shown in Fig. 13 for various gains C_2 . A value of $C_2 \leq 0.001$ maintains the maximum deviation to less than 2 (naut mi/hr)^2 out of the $7033989.7388 \text{ (naut mi/hr)}^2$ initial value. The discontinuity in the curves at approximately 38.9 hr is due to a computational inaccuracy which occurs as a result of the fact that the computing interval spans the apocenter relative to the virtual mass. As noted, the discussion of the time of flight in Chapter III Section A, such an occurrence

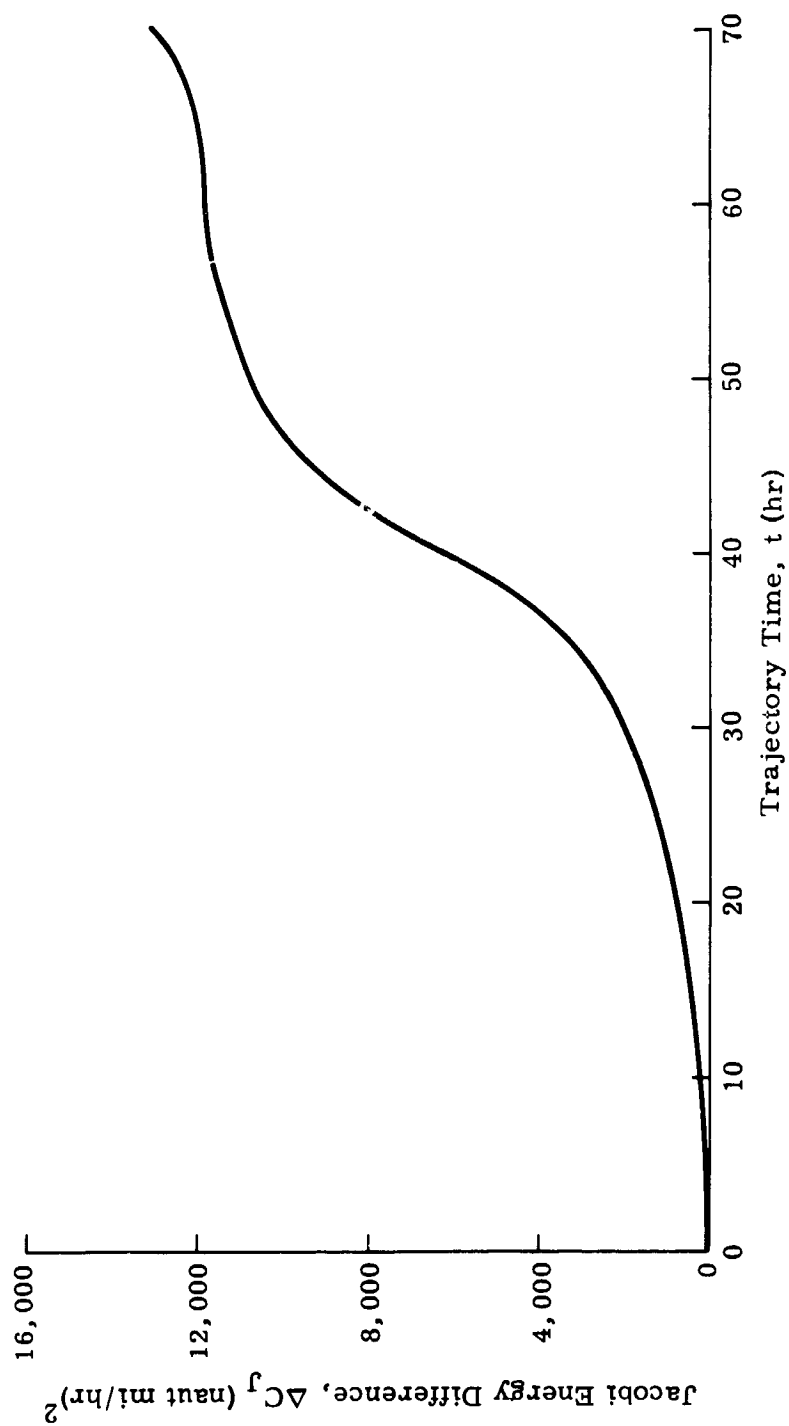


Fig. 11. Variation of Jacobi Energy Due to Change of Virtual Mass Interpolated Magnitude away from Midpoint by $\Delta C_1 = 0.1$ ($C_J = 7033989.7$)

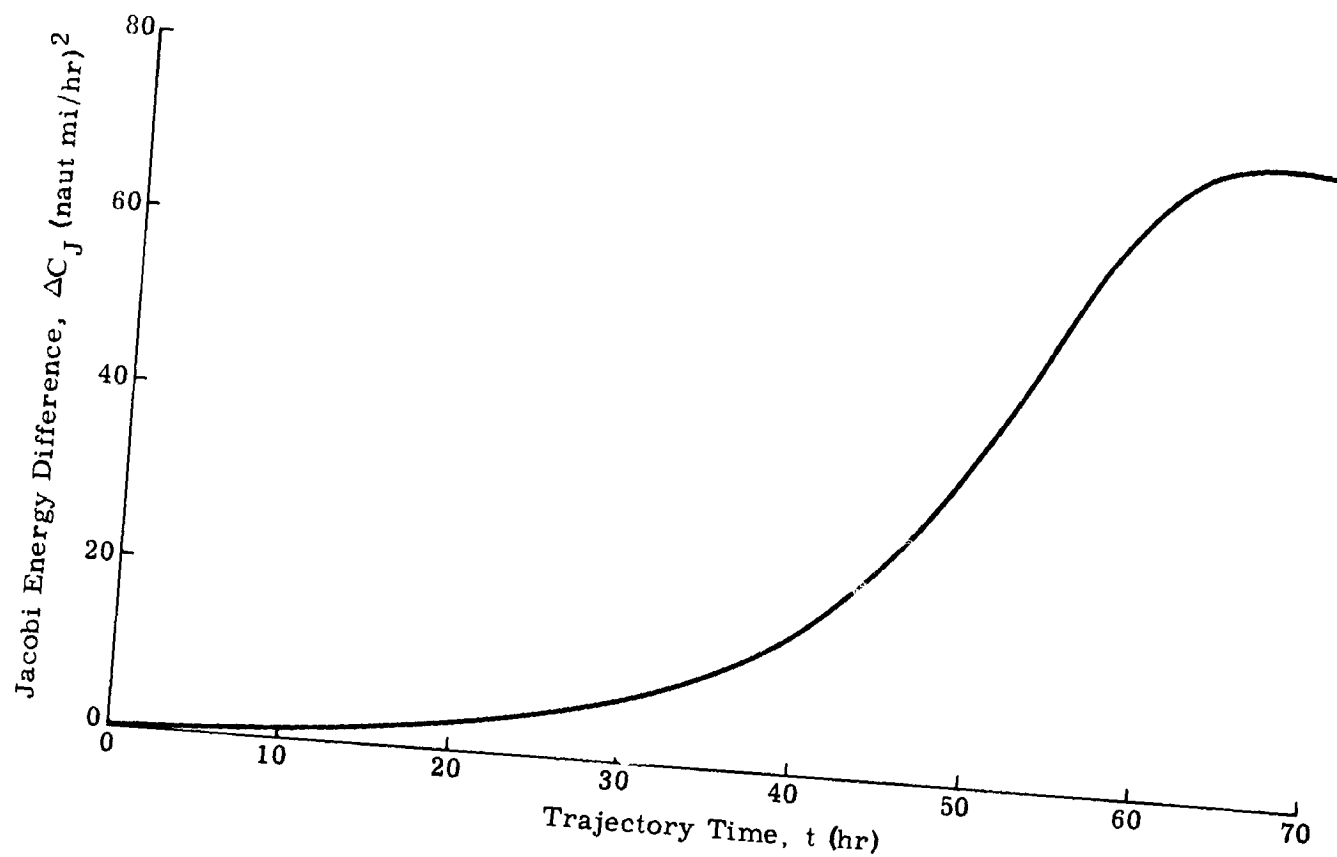


Fig. 12. Variation of Jacobi Energy Due to Time Bias of $|\Delta C_3| = 0.00001$ ($C_J = 7033989.7$)

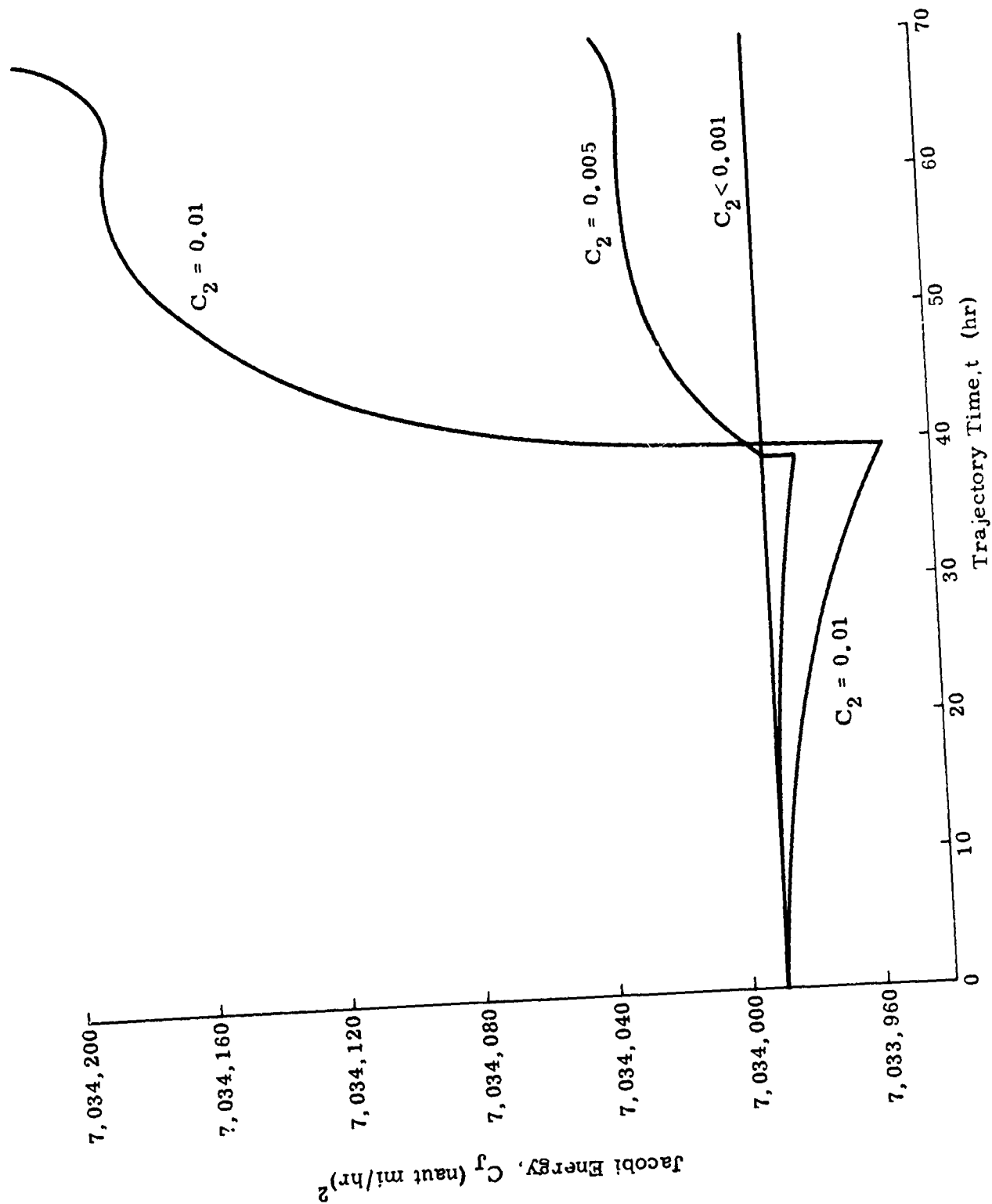


Fig. 13. Jacobi Energy Variation with Computation Interval Gain C_2

would cause the formulas to compute a large negative value for the time of flight. The program was provided with the capability to theoretically correct this mistake by adding a time equal to one period of the orbit. Unfortunately, some loss of significance occurs due to the subtraction of two nearly equal numbers. Apparently, the resulting inaccuracy is enough to cause the Jacobi energy variation indicated. This difficulty should be less serious for a machine such as the CDC 3600 than as shown here for the IBM 7094. The former carries 20 digits in double precision and also has automatic rounding, whereas the latter carries only 16 digits and simply truncates.

To gain some insight into what these variations in the Jacobi energy mean in terms of the spacecraft positional deviations, the differences Δr were computed between the base run with $C_2 = 0.001$ and other trajectories run with gains $C_2 > 0.001$. These differences, divided by the magnitude of the spacecraft position vector from the barycenter, are shown plotted in Fig. 14. They show, as an example, that a gain of $C_2 = 0.005$ gives a positional displacement of $\Delta r = 0.307$ naut mi for a total position vector distance of $r = 197299.51$ naut mi at 65.0 hr (just prior to the pericynthion at 70.338787 hr). Figure 13 shows that this same trajectory showed a Jacobi energy variation of $\Delta C_J = 44.66$ (naut mi/hr)² at this time.

An independent check of this Jacobi energy versus position deviation correspondence was made by comparing the same base run with identically the same trajectory numerically integrated by a standard procedure in an entirely different computer program. The integration tolerance happened to be set in that program so that the resulting solution displayed a Jacobi energy variation at 65 hr which was very nearly the same as that noted above (for the $C_2 = 0.005$ case). The positional difference between the numerically integrated trajectory and the base run was also approximately the same $\Delta r = 0.384$ naut mi. Thus, it is concluded that the Jacobi energy does provide a good index to accuracy and that the base run apparently is an accurate solution to the problem.

Of course, the price of accuracy is computation time. The run with gain $C_2 = 0.005$ calculated the trajectory in 2369 increments and took 57 sec on the computer. This time was measured for a complete problem cycle, from the time the instruction was given to read the input data until the program again sought data for the next problem.

Program accuracy control by means of the constant C_2 would require some study on the part of the user to determine what value to use to obtain a certain accuracy and to estimate the expected running time on the computer. The problem has been simplified somewhat by utilizing some

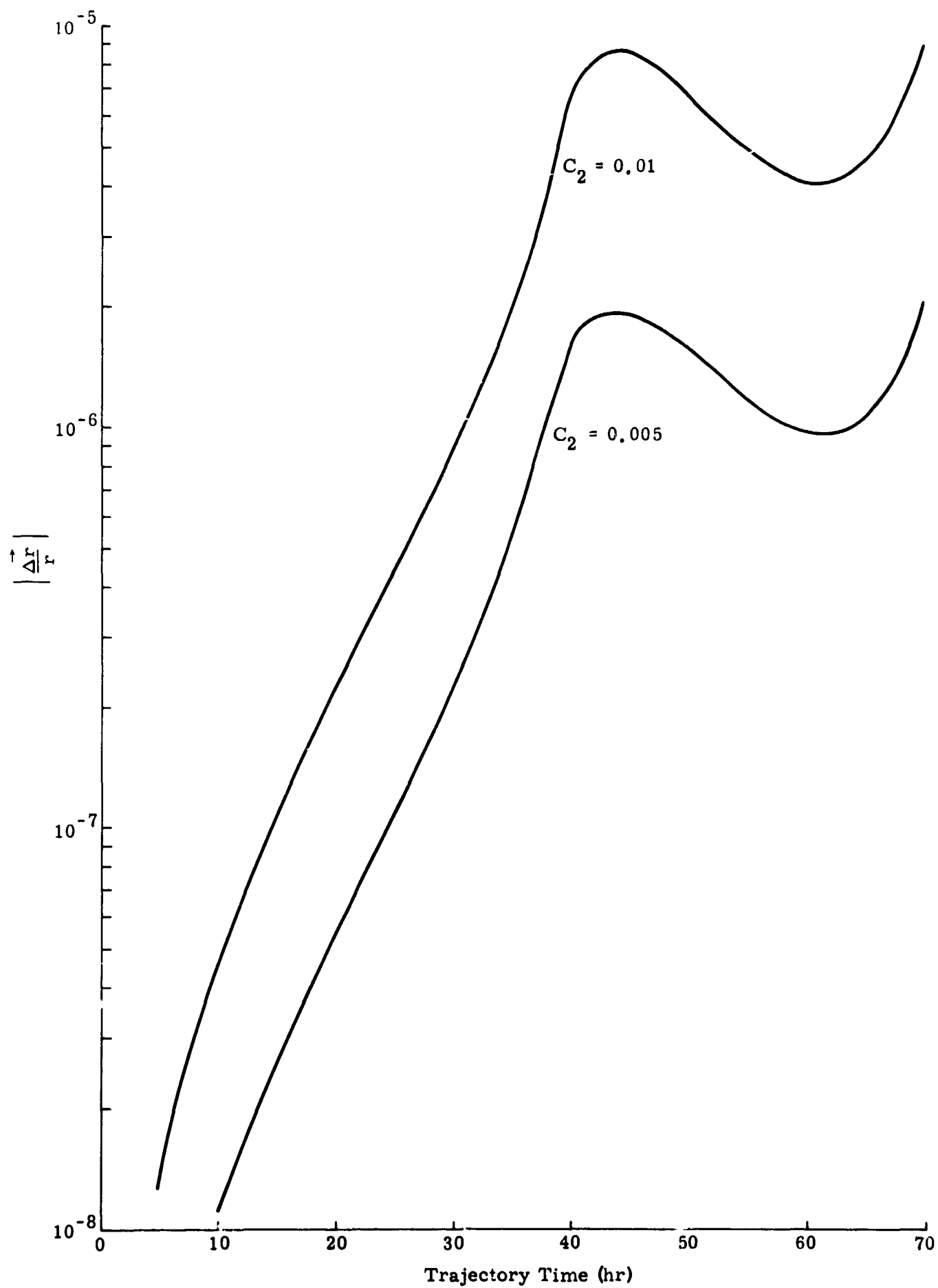


Fig. 14. Spacecraft Position Vector Error as a Function of Time for Various Computing Step Size Gains, C_2

information obtained from a cross-plot of Fig. 14. Since the maximum error occurs at the 70-hr point near pericynthion, a plot of C_2 versus $\Delta r/r$ at 70 hr was made as shown in Fig. 15. This curve was approximated by a second degree polynomial fit and built into the initialization section of the program. Thus, the user can input the more intuitively meaningful number $\Delta r/r$, or fractional accuracy desired at pericynthion, and the program will internally set its own gain appropriately.

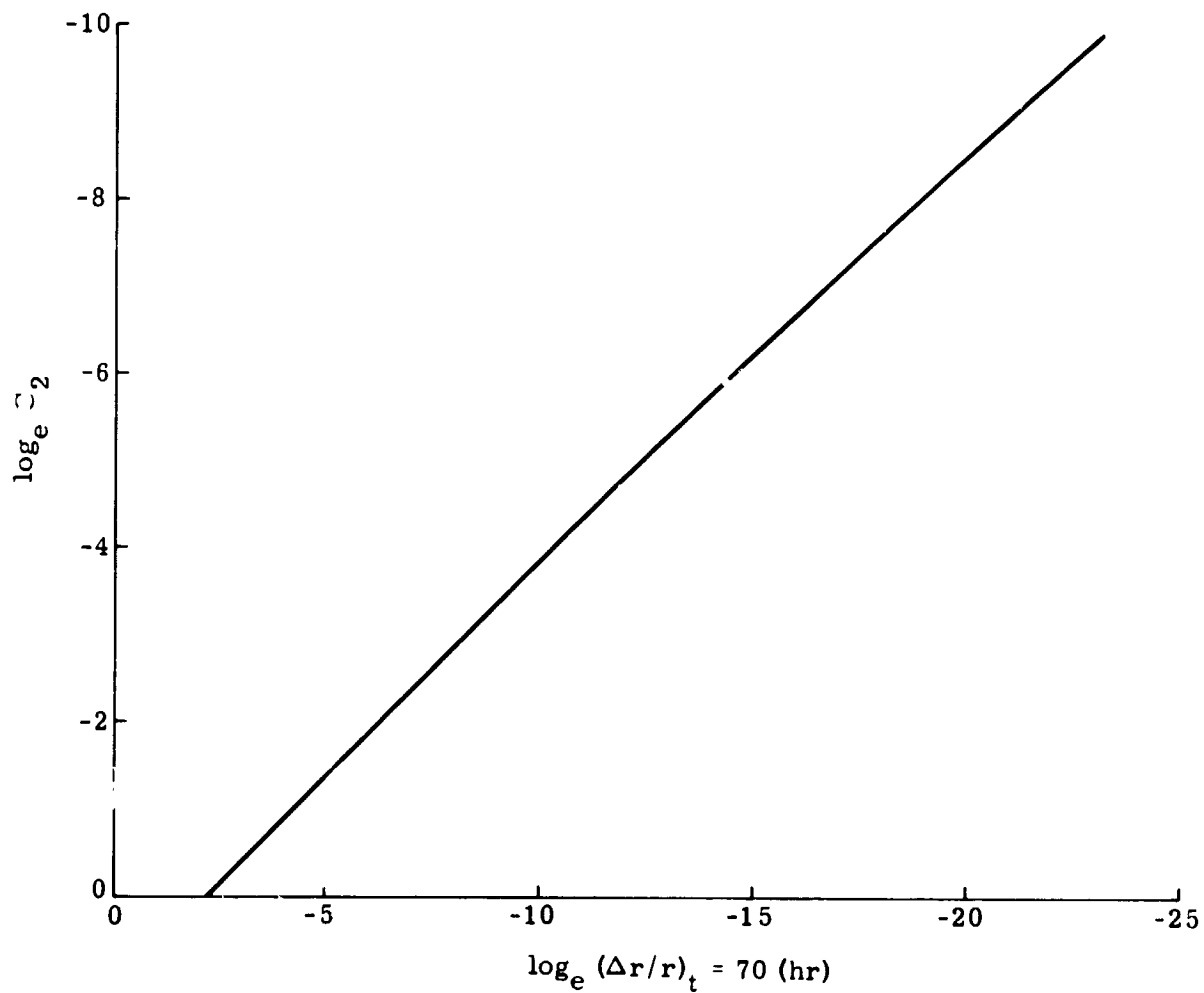


Fig. 15. Correlation of Computing Interval Gain C_2 with Position Error at $t = 70 \text{ (hr)}$

BLANK PAGE

V. GENERAL DESCRIPTION AND USE OF THE VIRTUAL MASS PROGRAM FOR COMPUTING SPACE TRAJECTORIES

This chapter is intended to serve two purposes.

- (1) It can be used, independently of the rest of this report, by the mission analyst. Usually he is not so much concerned about the computation process or the implementation of the procedure. Instead he is more interested in what general problem is solved, what is the solution accuracy and how can he use this digital computer program to calculate trajectories.
- (2) It also provides a broad overview of the digital program and its use, uncluttered by details. Thus it serves as an introduction to the trajectory analyst who may be interested in the details given in Chapter VI.

A. GENERAL DESCRIPTION

The purpose of this contract was to investigate the feasibility of the technique of computing space trajectories as a series of conic section solutions relative to a moving and varying virtual mass. At every instant of time this virtual mass replaces the combined effects of all the gravitating bodies upon the spacecraft. As explained in the preceding chapter, this was done by testing the procedure in a digital computer program. The simplest n -body problem, the restricted three-body problem, was used for this purpose. The final version of this FORTRAN IV source program is delivered with this report in fulfillment of the obligations under the contract.

Although this program solves only the restricted three-body problem, the virtual mass subroutine was formulated in completely general form. Thus, the only changes required in the program to make it capable of computing trajectories for more than two gravitating bodies are to replace the ephemeris and the input and output subroutines.

The reference coordinate system is a set of inertially oriented axes, centered at the barycenter. The xy plane is the earth-moon orbital plane and the initial position of the moon (and hence the earth) is given in terms of the ephemeris time t_{eph} , or time since the moon crossed the positive x -axis. Specification of the earth-moon distance D , the angular rate ω and the ratio of moon mass to total system mass μ complete the description of the gravitational environment. The total

mass of the system (times the Universal Gravitation Constant) is computed internally in the program as

$$\mu_e + \mu_m = \omega^2 D^3 \quad (V-1)$$

to ensure dynamical consistency of the system. The spacecraft initial position and velocity components are specified in the reference barycentric coordinate system.

To permit the greatest freedom possible in the use of the program and to scale numbers for greatest computational accuracy, all calculations within the program are carried out using dimensionless quantities. Thus, no units are specified (except that ω is presumed given in degrees per unit time) and the user may input the necessary data in any system of units he chooses. The value of ω^{-1} is chosen as the unit of time and D as the unit of length. Thus, conversion factors

$$\begin{aligned} \omega^{-1} &= \text{time} \\ D &= \text{length} \\ \omega D &= \text{velocity} \\ \omega^2 D^3 &= \text{mass times Universal Gravitation Constant} \\ \omega^3 D^3 &= \text{mass rate} \\ \text{etc.} \end{aligned}$$

are used to nondimensionalize input data and to convert dimensionless computed values to output data in the same system of units as the input.

The program has been discussed here in terms of the earth-moon system but, of course, is applicable to any two bodies (e.g., sun and a planet). The input constants determine the details of the system. It is only necessary to remember that the ephemeris time gives the time elapsed since the body designated by mass ratio μ ($0 < \mu < 1$) crossed the x-axis.

As described in Chapter IV, Section B, the accuracy of the solution is controlled by an input number $\Delta r/r$. This quantity represents the allowable error (nondimensionalized by D) in spacecraft position at the pericynthion point for an earth-moon trajectory. The program automatically converts this number to an equivalent gain controlling the computing interval size to maintain roughly equal steps in true anomaly relative to the virtual mass (see Chapter III, Section C). The gain correspondence was established for the lunar trajectory and it is not known at present what performance could be expected for an

interplanetary trajectory (i. e. , one which does not pass close to the large body).

The computing interval is adjusted as a print time is approached to cause the printout to occur exactly at a specified time increment. A similar adjustment is made whenever a major axis (pericenter or apocenter) crossing is imminent. The simple logic for this latter adjustment is not adequate for all cases. The pericenter crossings are picked up rather consistently except when such a crossing occurs shortly after the initial point (this is the case for the test trajectory selected) and a very loose gain is used. In this situation the program steps over this region before it has a chance to anticipate it. The apocenter crossing (which occurs at about 38.9 hr for the test trajectory) is only rarely caught by the routine provided.

Three stopping conditions are specified as input data, and the problem will terminate on whichever condition is met first. The conditions are a maximum allowable trajectory time and an impact with either of the two gravitating bodies. The radii of the two bodies must be given.

B. INPUT AND OUTPUT

A series of from 2 to 7 cards (1 to 6 data cards and 1 problem card) is used to input the data for a given problem. A number of problems may be run consecutively by inputting a sequence of such series of cards. Formats for the 7 cards are given below. The control word on each card begins in Column 1 and must appear exactly as specified. The variables begin in the columns indicated and must be punched according to the standard FORTRAN formats supplied (D18.0 indicates a double precision numeric field of 18 columns and I1 indicates an integer field of 1 column).

Data Card 1--

Col. 1	Col. 9 (D18.0)	Col. 27 (D18.0)	Col. 45 (D18.0)	Col. 63 (D18.0)
POSITION	t	x_s	y_s	z_s

Data Card 2--

Col. 1	Col. 9 (D18.0)	Col. 27 (D18.0)	Col. 45 (D18.0)
VELOCITY	\dot{x}_s	\dot{y}_s	\dot{z}_s

Data Card 3--

Col. 1	Col. 9 (D18. 0)	Col. 27 (D18. 0)	Col. 45 (D18. 0)	Col. 63 (D18. 0)
EFEMERIS	t_{eph}	ω	D	μ

Data Card 4--

Col. 1	Col. 9 (D18. 0)
ACCURACY	$\Delta r/r$

Data Card 5--

Col. 1	Col. 9 (D18. 0)	Col. 27 (D18. 0)	Col. 45 (D18. 0)
STOP	t_f	r_{1s_F}	r_{2s_F}

Data Card 6--

Col. 1	Col. 9 (D18. 0)	Col. 42 (I 1)	Col. 43 (I 1)	Col. 44 (I 1)
PRINT	Δt_p	IPRT1	IPRT2	IPRT3

IPRT1, IPRT2, and IPRT3 are used to indicate printout option requests. Ordinarily (and in case Columns 42 to 44 of this card are left blank), only the standard block of printout is given (see discussion below on output). However, any or all of the optional printout blocks may be obtained at each print interval by using the integers 1, 2, and 3 in Columns 42 to 44. A 1 appearing anywhere in these columns would request the first optional block in addition to the standard block of output. A 2 would request the second optional block, and a 3, the third. Thus, any combination of the integers 1, 2, and 3 may appear in any of the three columns to request any combination of the 3 optional blocks in addition to the standard block.

Problem Card--

Col. 1	Col. 9 (D18. 0)
PROBLEM	NPROB

NPROB is the problem number and will be truncated to an integer before being stored by the program. This number is used to identify the output.

To make the program as convenient to use as possible, certain flexibilities of the input have been incorporated in the program:

- (1) For any problem, the six data cards may appear in any order. The problem card, however, must always appear last.
- (2) On the first problem of a job, the data cards PRINT and ACCURACY may be omitted. If the PRINT card is omitted, Δt_p is assumed to be 5., and only the standard block of output is given. If the ACCURACY card is omitted, $\Delta r/r$ is assumed to be 1. D-7.
- (3) On any problem after the first problem of a job, any of the 6 data cards may be omitted. For those cards which do not appear in a given problem, the variables used in the first problem are always assumed. The problem card can never be omitted and must always be the last input card for a problem.
- (4) Any of the variable fields, if left blank, are assumed to be 0. D0.

For each problem, the input data are printed out as the first page of output. The sequence of fields corresponds to those on the input cards, but the cards are ordered in a standard sequence and any assumed cards (by omission assumed same as first problem) are also printed.

Subsequent pages of output for each problem give the standard block of output, followed by any optional blocks requested, at each printing interval. The optional blocks are always ordered 1, 2, and 3 if they appear. (See the PRINT data card for the method of requesting optional blocks of output.) All variables are dimensioned in the same units as the input.

Standard output block (option 0)

TRAJECTORY TIME = t

SPACECRAFT INERTIAL TRAJECTORY

POSITION.	x_s	y_s	z_s	r_s
VELOCITY	\dot{x}_s	\dot{y}_s	\dot{z}_s	\dot{r}_s

Optional output block 1

EPHEMERIS TIME = t_{eph}

EPHEMERIS DATA

POSITION OF EARTH	x_E	y_E	z_E	r_E
VELOCITY OF EARTH	\dot{x}_E	\dot{y}_E	\dot{z}_E	\dot{r}_E
POSITION OF MOON	x_M	y_M	z_M	r_M
VELOCITY OF MOON	\dot{x}_M	\dot{y}_M	\dot{z}_M	\dot{r}_M

Optional output block 2

SPACECRAFT RELATIVE TRAJECTORIES

POSITION REL TO EARTH	x_{Es}	y_{Es}	z_{Es}	r_{Es}
VELOCITY REL TO EARTH	\dot{x}_{Es}	\dot{y}_{Es}	\dot{z}_{Es}	\dot{r}_{Es}
POSITION REL TO MOON	x_{Ms}	y_{Ms}	z_{Ms}	r_{Ms}
VELOCITY REL TO MOON	\dot{x}_{Ms}	\dot{y}_{Ms}	\dot{z}_{Ms}	\dot{r}_{Ms}

Optional output block 3

VIRTUAL MASS DATA

VIRTUAL MASS POSITION	x_v	y_v	z_v	r_v
VIRTUAL MASS VELOCITY	\dot{x}_v	\dot{y}_v	\dot{z}_v	\dot{r}_v
SPACECRAFT POS REL TO VM. . . .	x_{vs}	y_{vs}	z_{vs}	r_{vs}
SPACECRAFT VEL REL TO VM	\dot{x}_{vs}	\dot{y}_{vs}	\dot{z}_{vs}	\dot{r}_{vs}
KEPLER (ANG MOM.) VECTOR	k_x	k_y	k_z	k
ECCENTRICITY VECTOR.	e_x	e_y	e_z	e
VM MAGN	$= \mu_v$			
VM MAGN RATE	$= \dot{\mu}_v$			

C. SAMPLE PROBLEM

The foregoing descriptions of certain features of the program and of the input and output formats are best illustrated by an example. This section, therefore, lists the output of a sample problem. As noted in Chapter IV, Section A, this example is the problem used as a reference for the digital computer study reported there.

Observe that all output options were requested. Distances are expressed in terms of nautical miles and times are in hours. In addition to the printout at the requested 5-hr increments, there are also outputs at $t = 0.002900911 \dots$, $t = 70.338748577$ and $t = 70.4000645948$ hr. The first occurred because of the fact that the trajectory insertion was made with a slightly negative flight path angle. Since the virtual

mass almost exactly coincides with the location of the earth, periapsis relative to the fictitious body was achieved almost immediately. The second time corresponds to the periapsis passage at pericyynthion. Both of these occurrences were preceded by the notation "MAJOR AXIS CROSSING" in the printout. The last time point was printed as one of the stopping conditions (maximum time $t = 70.4$ hr) was met.

For the ACCURACY used in this example, the Jacobi energy variation was less than 2 parts out of $C_J = 7033989.7$ (naut mi/hr)².

V I T A L	M A S S	P R O G R A M	F L R	C O M P U T I N G	S P A C E	I R A J E C T O R I E S	P A G E
PRINT	5.0000000000000000	1	2	3			1
POSITION	0.0000000000000000	-1.1260880000000000	03	-5.4330951000000000	03	1.9597270000000000	02
VELOCITY	1.8364875000000000	3.1525321000000000	03	1.0624849000000000	04		
ELEMENTS	9.5591177000000000	01	5.4901450000000000	01	2.0774720000000000	05	1.2143289000000000
ACCURACY	1.0000000000000000	07					
STOP	7.0000000000000000	01	3.4440000000000000	03	9.2800000000000000	02	
PROBLEM	1						

X - CUM. Y - CUM. Z - CUM. RESULTS

TRAJECTORY TIME = 2.50051145148E-03
 SPACECRAFT INITIAL TRAJECTORY
 POSITION -1.072050430770 03 -5.423672944540 03 2.267780494890 02 5.533413461850 03
 VELOCITY 1.833865932000 04 3.343384772040 03 1.061322035440 04 2.145052665270 04

EFFECTIVE TIME = 5.25540775115E-01
 EFFECTIVE DATA
 POSITION OF EARTH -1.574414150440 03 -1.971143912330 03 0.000000000000-39 2.52273288540 03
 VELOCITY OF EARTH 1.888773350020 01 -1.508622259310 01 0.000000000000-39 2.417313806710 01
 POSITION OF MOON 1.240786119580 05 1.603525817540 05 0.000000000000-39 2.052224465710 05
 VELOCITY OF MOON -1.536517387410 03 1.227264395350 03 0.000000000000-39 1.966485082050 03

SPACECRAFT RELATIVE TRAJECTORIES
 POSITION REL. TO EARTH 0.501563735670 03 -0.345252903220 04 0.226778049490 03 0.349613374790 04
 VELOCITY REL. TO EARTH 0.183197715860 05 0.335847099460 04 0.106132203540 05 0.214367396000 05
 POSITION REL. TO MOON -0.129151462430 06 -0.165776254700 06 0.226778049490 03 0.210147372810 06
 VELOCITY REL. TO MOON 0.198751767070 05 0.211612037670 04 0.106132203540 05 0.226305338180 05

VIRTUAL MASS DATA
 VIRTUAL MASS POSITION -1.574406827770 03 -1.971134724420 03 0.000000000000-39 2.522722529560 03
 VIRTUAL MASS VELOCITY 1.8889009951240 01 -1.508433232190 01 0.000000000000-39 2.417302550340 01
 SPACECRAFT POS. REL. TO V.M. 4.595136749250 02 -3.452839100720 03 2.258263736890 02 5.496142029280 03
 SPACECRAFT VEL. REL. TO V.M. 1.832062472240 04 3.352577311560 03 1.061360652680 04 2.143673735330 04
 KEPLER (ANG. MOM.) VECTOR -3.74041759210 07 -1.168606798220 06 6.493416863540 07 7.494587566950 07
 ECCENTRICITY VECTOR 1.399477112490-01 -9.6333509812620-01 6.327716827400-02 9.755175421320-01
 V.M. MAG. = 8.132527684770 11
 V.M. MAG. RATE = -5.18195525E74E-07

X = COMP. Y = COMP. Z = COMP. RESULTANT

TRAJECTORY TIME = 5.0000000000000000

SPACECRAFT INITIAL TRAJECTORY

POSITION	1.179067535980 04	3.515627912160 04	8.312586102870 03	3.800109307420 04
VELOCITY	3.60489095030 02	5.850641275050 03	3.069621494800 02	5.870008266270 03

EPOCHES TIME = 5.0551770000000000

EPOCHES DATA

POSITION OF EARTH	-1.478261817440 03	-2.044243207080 03	0.000700000000-39	2.522734288540 03
VELOCITY OF EARTH	1.958817998070 01	-1.416487942250 01	0.000000000000-39	2.417313806710 01
POSITION OF MOON	1.202566172120 05	1.662992102900 05	0.000000000000-39	2.052244657110 05
VELOCITY OF MOON	-1.593498684760 03	1.152313116990 03	0.000000000000-39	1.966485082050 03

SPACECRAFT RELATIVE TRAJECTORIES

POSITION REL. TO EARTH	0.1326889371770 05	0.372005223290 05	0.831258610290 04	0.403614004100 05
VELOCITY REL. TO EARTH	0.346460729520 03	0.586480615450 04	0.304962149480 03	0.588294043650 04
POSITION REL. TO MOON	-0.108465941850 06	-0.131142931170 06	0.831258610290 04	0.170389049020 06
VELOCITY REL. TO MOON	0.195954759430 04	0.469832815810 04	0.304962149480 03	0.509971726350 04

VIRTUAL MASS DATA

VIRTUAL MASS POSITION	-1.458375237740 03	-2.016742662190 03	0.000000000000-39	2.488796757380 03
VIRTUAL MASS VELOCITY	2.527203721280 01	-2.134081355180 01	0.000000000000-39	2.927285200840 01
SPACECRAFT PLS. REL. TO V.M.	1.324626151210 04	3.712500315910 04	8.100805971 03	4.028393402300 04
SPACECRAFT VEL. REL. TO V.M.	3.381359687920 02	5.854655733140 03	3.05807143640 02	5.872380070810 03
KEPLER (ANG. MOM.) VECTOR	-3.727956983100 07	-1.240870487030 06	6.499912143060 07	7.495120717010 07
ECCENTRICITY VECTOR	1.404122954500-01	-9.627277337810-01	6.231096081580-02	5.7493354660010-01
V.M. MAGN.	8.114535647360 11			
V.M. MAGN. RATE	-6.26171218420 08			

X = COMP Y = COMP Z = COMP RESULTANT

TRAJECTORY TIME = 1.000000000 01
 SPACECRAFT INERTIAL TRAJECTORY
 POSITION 1.235351910870 04 5.026436034230 04 9.030711625280 03 6.217682364880 04
 VELOCITY -4.754039296670 01 4.405446144030 03 3.986024403150 01 4.405382959860 03

EPIHEMERIS TIME = 1.035511770000 02
 EPIHEMERIS DATA
 POSITION OF EARTH -1.378662087720 03 -2.112694757570 03 0.000000000000 -39 2.522714288540 03
 VELOCITY OF EARTH 2.024409082610 01 -1.321050304250 01 0.000000000000 -39 2.417313806710 01
 POSITION OF MOON 1.121541779620 05 1.718677447690 05 0.000000000000 -39 2.052244657110 05
 VELOCITY OF MOON -1.646857040180 03 1.074674586620 03 0.000000000000 -39 1.966445082050 03

SPACECRAFT RELATIVE TRAJECTORIES
 POSITION REL. TO EARTH 0.137321811960 05 0.623770641000 05 0.903071162530 04 0.645060049810 05
 VELOCITY REL. TO EARTH -0.677844837930 02 0.441865664710 04 0.398602440310 02 0.641935637380 04
 POSITION REL. TO MOON -0.558006588540 05 -0.111603375430 06 0.903071162530 04 0.149990128560 06
 VELOCITY REL. TO MOON 0.159931664720 04 0.333077155740 04 0.398602440310 02 0.369503639260 04

VIRIAL MASS DATA
 VIRTUAL MASS POSITION -1.267756507150 03 -1.942741006230 03 0.000000000000 -39 2.319795291170 03
 VIRTUAL MASS VELOCITY 4.843237302320 01 3.354417963010 01 0.000000000000 -39 5.831440183610 01
 SPACECRAFT PCS. REL. TO V.M. 1.362280747930 04 6.213705192060 04 9.030069590640 03 6.425057401930 04
 SPACECRAFT VEL. REL. TO V.M. -5.527197793460 01 4.375000681410 03 4.029265921810 01 4.376222460320 03
 KEPLER (ANG. MOM.) VECTOR -3.700245859500 07 -1.409307089800 06 6.551971184410 07 7.525955573000 07
 ECCENTRICITY VECTOR 1.432652700230 -01 -9.612174629080 -01 6.023403531170 -02 9.737002041710 -01
 V.M. MAGN. = 8.06549066370 11
 V.M. D. RATE = -1.212250610610 09

X - COMP. Y - COMP. Z - COMP. RESULTANT

TRAJECTORY TIME = 2.5000000000000000
 SPACECRAFT INERTIAL TRAJECTORY
 POSITION 9.320542801010 03 1.126517076380 05 8.016590201140 03 1.133205430200 05
 VELOCITY -2.789134409520 02 2.869478425120 03 -1.255196686550 02 2.885732892530 03

EPIHEMERIS TIME = 1.1855117000000000
 EPIHEMERIS DATA
 POSITION OF EARTH -1.061828925510 03 -2.288385374790 03 0.000000000000 -39 2.522734288854 03
 VELOCITY OF EARTH 2.192757908220 01 -1.017457024170 01 0.000000000000 -39 2.417213806710 01
 POSITION OF MOON 8.637579628040 04 1.861601786660 05 0.000000000000 -39 2.052244657110 05
 VELOCITY OF MOON -1.783808830740 03 8.277014155550 02 0.000000000000 -39 1.966485082050 03

SPACECRAFT RELATIVE TRAJECTORIES
 POSITION REL. TO EARTH 0.103623717270 05 0.114940093010 06 0.801659020110 04 0.115686145860 06
 VELOCITY REL. TO EARTH -0.300841020030 03 0.287965299540 04 -0.125519668660 03 0.289804449250 04
 POSITION REL. TO MOON -0.770592534790 05 -0.735084710290 05 0.801659020110 04 0.106798359440 06
 VELOCITY REL. TO MOON 0.150489538980 04 0.20417700960 04 -0.125519668660 03 0.253955508812 04

VIRIAL MASS DATA
 VIRTUAL MASS POSITION 2.833434766730 02 6.106436286340 02 0.000000000000 -39 6.731784064900 02
 VIRTUAL MASS VELOCITY 1.842263596020 02 4.123580842540 02 0.000000000000 -39 4.516398357340 02
 SPACECRAFT PCS. REL. TO V.M. 9.046571510270 03 1.119912960770 05 8.019130484730 03 1.126418985400 05
 SPACECRAFT VEL. REL. TO V.M. -4.628867322450 02 2.458873674420 03 -1.254330118220 02 2.505206042240 03
 KEPLER (ANG. MOM.) VECTOR -3.376543440530 07 -2.577210394320 06 7.408366161110 07 8.145634101600 07
 ECCENTRICITY VECTOR 1.578545316140 -01 -9.437617561980 -01 3.911480167320 -02 9.576713421660 -01
 V.M. MAGN. = 7.634361827070 11
 V.M. MAGN. RATE = -5.536250448470 05

VIRTUAL MASS PROGRAM FOR COMPUTING SPACE TRAJECTORIES									
PROBLEM 1 PAGE 9									
X - COMP. Y - COMP. Z - COMP. RESULTANT									
TRAJECTORY TIME = 3.000000000000 C1									
SPACECRAFT INERTIAL TRAJECTORY									
POSITION	7.876525292130 03	1.263070203590 05	7.342268014970 03	1.267651845150 05					
VELOCITY	-2.967791381330 02	2.602574573980 03	-1.430347359300 02	2.623334358440 03					
EPIHEMERIS TIME = 1.235611770000 C2									
EPIHEMERIS DATA									
POSITION OF EARTH	-9.510145284530 02	-2.336612859940 03	0.000000000000 -39	2.522734288540 03					
VELOCITY OF EARTH	2.238970054410 01	-9.112733594070 00	0.000000000000 -39	2.417313806710 01					
POSITION OF MOON	7.736504370360 04	1.900834851830 05	0.000000000000 -39	2.052244657110 05					
VELOCITY OF MOON	-1.821402417400 03	7.413209910810 02	0.000000000000 -39	1.966485082050 03					
SPACECRAFT RELATIVE TRAJECTORIES									
POSITION REL. TO EARTH	0.882754382060 04	0.128643633220 06	0.734226801500 04	0.129155018470 06					
VELOCITY REL. TO EARTH	-0.319169438680 03	0.261168730760 04	-0.143034735930 03	0.263500259180 04					
POSITION REL. TO MOON	-0.694885144110 05	-0.637764648240 05	0.734226801500 04	0.946044396430 05					
VELOCITY REL. TO MOON	0.152462267930 04	0.186125358290 04	-0.143034735930 03	0.241022782110 04					
VIRTUAL MASS DATA									
VIRTUAL MASS POSITION	1.424275614290 03	3.499400500080 03	0.000000000000 -39	3.778143047240 03					
VIRTUAL MASS VELOCITY	2.782849595200 02	7.797713387740 02	0.000000000000 -39	8.279407462680 02					
SPACECRAFT POS. REL. TO V.M.	6.489595675300 03	1.226876158540 05	7.351653901680 03	1.230769075170 05					
SPACECRAFT VEL. REL. TO V.M.	-5.741535721340 02	1.823152235760 03	-1.428465938020 02	1.922460879880 03					
KEPLER (ANG. MOM.) VECTOR	-3.097280219730 07	-3.293912655880 06	8.231270622450 07	8.800878332610 07					
ECCENTRICITY VELOCITY	1.533937330530 -01	-9.258415732100 -01	2.066986440930 -02	9.386903106560 -01					
V.M. MAGN.	7.278706890710 11								
V.M. MAGN. RATE	-8.962501654700 05								

X - COMP. Y - COMP. Z - COMP. RESULTANT

TRAJECTORY TIME = 4.500000000000000001

SPACECRAFT INERTIAL TRAJECTORY

POSITION	3.306163505110 03	1.608374017700 05	4.962265976390 03	1.609478940800 05
VELOCITY	-3.031994522270 02	2.048296039760 03	-1.701662794310 02	2.077595518140 03

EPIHEMERIS TIME = 1.365511770000 02

EPIHEMERIS DATA

POSITION OF EARTH	-0.065176684510 02	-2.448739391690 03	0.000000000000 00	0.000000000000 00	2.522734288540 03
VELOCITY OF EARTH	2.346411022150 01	-5.811723971970 00	0.000000000000 00	0.000000000000 00	2.417313806710 01
POSITION OF MOON	4.534021986360 04	1.992049799340 05	0.000000000000 00	0.000000000000 00	2.052244657110 05
VELOCITY OF MOON	-1.908800565800 03	4.727838173160 02	0.000000000000 00	0.000000000000 00	1.966485082050 03

SPACECRAFT RELATIVE TRAJECTORIES

POSITION REL. TO EARTH	0.391268117360 04	0.163286141160 04	0.496226597640 04	0.163408375100 06
VELOCITY REL. TO EARTH	-0.326663562450 03	0.205410776370 04	-0.170166279430 03	0.208686950980 04
POSITION REL. TO MOON	-0.460340563580 05	-0.383675701640 05	0.496226597640 04	0.601317676650 05
VELOCITY REL. TO MOON	0.160560620580 04	0.157551222240 04	-0.170166279430 03	0.225591131000 04

VIRTUAL MASS DATA

VIRTUAL MASS POSITION	9.276775581870 03	3.745382365070 04	0.000000000000 00	3.858558631480 04
VIRTUAL MASS VELOCITY	8.272638457300 02	4.877827020780 03	0.000000000000 00	4.947480360250 03
SPACECRAFT POS. REL. TO V.M.	-5.532075595880 03	1.234797522290 05	4.968069429420 03	1.237215481840 05
SPACECRAFT VEL. REL. TO V.M.	-1.129842418840 03	-2.819161205080 03	-1.701269231050 02	3.041910222160 03
KEPLER (ANG. MOM.) VECTOR	-7.001441724790 06	-6.622341349860 06	1.562361394010 08	1.565330854670 08
ECCENTRICITY VECTOR	-9.565537001230 01	-5.937837348850 01	-6.803479405750 02	1.127919694940 00
V.M. MAGN. =	4.390405121810 11			
V.M. MAGN. RATE =	-3.301824376540 10			

X - COMP. Y - COMP. Z - COMP. RESULTANT

TRAJECTORY TIME = 5.5000000000000001

SPACECRAFT INERTIAL TRAJECTORY

POSITION	3.543005612040 02	1.800280873680 05	3.208299946570 03	1.800571045620 05
VELOCITY	-2.720650139330 02	1.803903972350 03	-1.803995503700 02	1.833292647230 03

EPIPHORES TIME = 1.4855117000000000

EPIPHORES DATA

POSITION OF EARTH	-3.69431672930 02	-2.495534541490 03	0.000000000000-39	2.522734288540 03
VELOCITY OF EARTH	2.391250687660 01	-3.540143907690 00	0.000000000000-39	2.417313806710 01

POSITION OF MOUN	3.005501974880 04	2.030117659510 05	0.000000000000-39	2.052244657110 05
VELOCITY OF MOUN	-1.945282731470 03	2.879907508680 02	0.000000000000-39	1.966485082050 03

SPACECRAFT RELATIVE TRAJECTORIES

POSITION REL. TO EARTH	0.763754128500 03	0.182523621910 06	0.3208299954660 04	0.182553414280 06
VELOCITY REL. TO EARTH	-0.296581520810 03	0.180744411630 04	-0.180399550370 03	0.184047788080 04

POSITION REL. TO MOUN	-0.296007187880 05	-0.229836785820 05	0.320829994660 04	0.376603360140 05
VELOCITY REL. TO MOUN	0.167261371750 04	0.151591322150 04	-0.180399550370 03	0.226454709400 04

VIRTUAL MASS DATA

VIRTUAL MASS POSITION	1.737869492180 04	1.173873640900 05	0.000000000000-39	1.186668120630 05
VIRTUAL MASS VELOCITY	4.242576986940 02	1.063003195330 04	-0.000000000000-39	1.063849490870 04

SPACECRAFT PCS. REL. TO V.M.	-1.698151335680 04	6.267718271720 04	3.209045176920 03	6.501614415050 04
SPACECRAFT VEL. REL. TO V.M.	-6.573171063460 02	-8.825670976100 03	-1.80395254510 02	8.85013370340 03

KEPLER (ANG. MOM.) VECTOR	1.701530779670 07	-5.301107267050 06	1.935791212500 08	1.3977845080 08
ECCENTRICITY VECTOR	-1.914459936380 01	5.335250228100-01	1.697391340930 00	1.922710251480 01

V.M. MAGN. = 8.803509360200 10

V.M. MAGN. RATE = -2.572551433040 10

X COMP. Y COMP. Z COMP. RESULTANI

TRAJECTORY TIME = 6.566666000000 01
 SPACECRAFT INERTIAL TRAJECTORY
 POSITION -1.828491442380 03 1.972865285500 05 1.333854642940 03 1.972995106340 05
 VELOCITY -1.245712926590 02 1.683359102390 03 -1.990772786130 02 1.699890733460 03

EPIHEMERIS TIME = 1.585911770000 02
 EPIHEMERIS DATA
 POSITION OF EARTH -1.289990531320 02 -2.519433971130 03 0.000000000000 -39 2.522734288540 03
 VELOCITY OF EARTH 2.414151403560 01 -1.236084171070 00 0.000000000000 -39 2.417313806710 01
 POSITION OF MOON 1.049407457480 04 2.049555848490 05 0.000000000000 -39 2.052244657110 05
 VELOCITY OF MOON -1.963912466700 03 1.005554626720 02 0.000000000000 -39 1.966485082050 03

SPACECRAFT RELATIVE TRAJECTORIES
 POSITION REL. TO EARTH -0.169949238930 04 0.195805962520 06 0.133385464290 04 0.199817642120 06
 VELOCITY REL. TO EARTH -0.148712806690 03 0.168482710810 04 -0.199077278610 03 0.170305303670 04
 POSITION REL. TO MOON -0.123225660170 05 -0.766945629920 04 0.133385464290 04 0.145755055270 05
 VELOCITY REL. TO MOON 0.183934117400 04 0.158303556130 04 -0.199077278610 03 0.243491464020 04

VIRTUAL MASS DATA
 VIRTUAL MASS POSITION 1.016892971200 04 1.936056978280 05 0.000000000000 -39 1.988658601710 05
 VIRTUAL MASS VELOCITY -1.738588279700 03 1.301929253380 03 -0.000000000000 -39 3.731865087900 03
 SPACECRAFT PCS. REL. TO V.M. -1.200717599760 04 -1.309369000860 03 1.335057800510 03 1.215191762440 04
 SPACECRAFT VEL. REL. TO V.M. 1.613625923770 03 -1.621480919760 03 -1.990504560610 02 2.296359835960 03
 KEPLER (ANG. MOM.) VECTOR 2.425401247070 06 -2.354776232400 05 2.158250500000 07 2.171963562220 07
 ECCENTRICITY VECTOR -4.890349675910 00 -5.816112107850 00 4.86111111755700 -01 7.614393214320 00
 V.M. MAGN. = 5.563405027340 05
 V.M. MAGN. RATE = 7.153070375540 08

X = COMP. Y = COMP. Z = COMP. RESULTANT

TRAJECTORY TIME = 7.033874857650 01

SPACECRAFT INITIAL TRAJECTORY

POSITION	8.25522346813D-02	2.06373023657D 05	1.33621192150D-02	2.06373023657D 05
VELOCITY	2.69325849722D 03	4.66115622573D-02	-5.04436194953D 02	2.74009073022D 03

EPIPHENIS TIME = 1.63525525577D 02

EPIPHENIS DATA

POSITION OF EARTH	-1.02962098692D-03	-2.52273428854D 03	0.0000000000D-39	2.52273428854D 03
VELOCITY OF EARTH	2.41731380671D 01	-9.86594996807D-06	0.0000000000D-39	2.41731380671D 01
POSITION OF MOON	8.37556800760D-02	2.05224465711D 05	0.0000000000D-39	2.05224465711D 05
VELOCITY OF MOON	-1.96648508205D 03	8.02595152462D-04	0.0000000000D-39	1.96648508205D 03

SPACECRAFT RELATIVE TRAJECTORIES

POSITION REL. TO EARTH	0.83581905667D-01	0.208895575795D 06	0.13362119215D-01	0.208895575795D 06
VELOCITY REL. TO EARTH	0.26690853592D 04	0.46621428207D-01	-0.50443619495D 03	0.27163343924D 04
POSITION REL. TO MOON	-0.12073953957D-02	0.11495579457D 04	0.13362119215D-01	0.11495579457D 04
VELOCITY REL. TO MOON	0.46597435793D 04	0.45803967105D-01	-0.50443619495D 03	0.46869676873D 04

V I R T U A L M A S S D A T A

VIRTUAL MASS POSITION	8.37585336061D-02	2.05221656679D 05	0.0000000000D-39	2.05221656679D 05
VIRTUAL MASS VELOCITY	-1.56645816557D 03	3.47403109109D-04	0.0000000000D-39	1.96645816557D 03
SPACECRAFT POS. REL. TO V.M.	-8.81242798553D-01	1.15136683611D 03	1.08630203821D-01	1.15136717848D 03
SPACECRAFT VEL. REL. TO V.M.	4.65571511284D 03	1.47097402218D 00	-5.04436118946D 02	4.68694060280D 03
KEPLER (ANG. MOM.) VECTOR	-5.80791178084D 05	6.16552139361D 01	-5.36504389432D 06	5.39638901343D 06
ECCENTRICITY VECTOR	-1.51785442138D-05	1.51153576275D 00	1.90137541074D-05	1.51153576265D 00
V.M. MAGN. =	1.0070553998D 10			
V.M. MAGN. RATE =	5.27152254732D 03			

STOPPING CONDITION--EXCEEDED MAXIMUM TRAJECTORY TIME

RESULTANT

Z - COMP.

Y - COMP

X - CLMP.

TRAJECTORY TIME = 7.641111455476 ul

SPACECRAFT INERTIAL TRAJECTILRY

VELOCITY	2.637164551460	03	-4.553767379250	02	-4.983689944670	02	2.722203672170	03
POSITION	1.640622545320	02	2.063589067210	05	-3.079122350340	01	2.063589742360	05
ACCELERATION	1.640622545320	02	2.063589067210	05	-3.079122350340	01	2.063589742360	05
INERTIAL TRAJECTORY	1.640622545320	02	2.063589067210	05	-3.079122350340	01	2.063589742360	05

[illegible]

EPFMERIS TIME = 1.629512415550 C2

REFERENCE DATA

MERIS DATA					
POSITION OF EARTH	• • • • •	1.481170857600 00	-2.522733853720 03	0.000000000000 00-39	2.522734286540 03
SPEED OF EARTH	• • • • •	2.417313390000 01	1.419275418910-02	0.000000000000 00-39	2.417313806710 01

PLSATION OF	2.417313806710	01
CITY OF	2.417313806710	01
EARTH	2.41927541891D-02	-39
APPLICATION	0.000000000000	00
2.417313806710	01	01

	PLSTITION CF MCCA	-1.20493267666D 02	2.052224430333D 05	0.000000000000=39	2.052224485711D 03	1.06448508205D 03
--	-------------------	--------------------	--------------------	-------------------	--------------------	-------------------

VELOCITY CF M/LUN	-1.96648474311D 03	-1.1545807296D 00	0.0000000000D -39	0.0000000000D -39
-------------------	---	---	---	---	---	---	---	--------------------	-------------------	-------------------	-------------------

SPACECRAFT RELATIVE TRAJECTORIES

[illegible]

VELOCITY REF. IC EAST	0.26129914176D 04	-0.45539093068D 03	-0.49836899447D 03	0.26987917116D 04
-----------------------	-------------------	--------------------	--------------------	-------------------

POSITION	REL.	TL	MCCN
0.113447638250	03	0.284555522200	03
-0.307912235030	04	0.113447638250	04
0.65277081430	02	0.1170022419080	02
0.1170022419080	04	0.65277081430	04

Variable	Mean	Std. Dev.	Minimum	Maximum
PERCENTAGE OF THE POPULATION WITH COLLEGE DEGREE	15.4603	6.6292	0.0000	30.0000
PERCENTAGE OF THE POPULATION WITH HIGH SCHOOL DEGREE	72.2222	15.7200	0.0000	100.0000
PERCENTAGE OF THE POPULATION WITH LESS THAN HIGH SCHOOL DEGREE	12.5720	3.6899	0.0000	44.7000
PERCENTAGE OF THE POPULATION WITH BACHELOR'S DEGREE	15.4603	6.6292	0.0000	30.0000
PERCENTAGE OF THE POPULATION WITH MASTER'S DEGREE	0.4652	0.7081	0.0000	4.3500
PERCENTAGE OF THE POPULATION WITH DOCTORAL DEGREE	0.4652	0.7081	0.0000	4.3500

VIRTUAL MASS DATA

VIRTUAL	MASS	POSITION							
-1.	20491523838D	02							
2.	05221460247D	05							
0.	00000000000D	-39							
2.	052214752619U	03							
0.	00000000000D	-39							
2.	052214743174U	03							
0.	00000000000D	-39							

[illegible]

SPACECRAFT	PCS.	REFL.	TL	V.M.	2.831252	262533D	02	1.1375854	339253D	03	-3.063657	167880	01	1.1726688	858260	01	1.1726688	858260	01	4.65255	958870D	03
SPACECRAFT	PCS.	REFL.	TL	V.M.	2.831252	262533D	02	1.1375854	339253D	03	-3.063657	167880	01	1.1726688	858260	01	1.1726688	858260	01	4.65255	958870D	03

[illegible]

KEPLER	(ANG.)	PLM.)	VECTUR.	PLM.)	VECTUR.	PLM.)	VECTUR.
1	51.0089532220	00	00	00	00	00	00
2	1.510989532220	00	00	00	00	00	00
3	3.942255841020	00	00	00	00	00	00
4	3.942255841020	00	00	00	00	00	00
5	3.942255841020	00	00	00	00	00	00
6	1.510989532220	00	00	00	00	00	00
7	51.0089532220	00	00	00	00	00	00
8	1.510989532220	00	00	00	00	00	00
9	3.942255841020	00	00	00	00	00	00
10	3.942255841020	00	00	00	00	00	00
11	3.942255841020	00	00	00	00	00	00
12	1.510989532220	00	00	00	00	00	00
13	51.0089532220	00	00	00	00	00	00
14	1.510989532220	00	00	00	00	00	00
15	3.942255841020	00	00	00	00	00	00
16	3.942255841020	00	00	00	00	00	00
17	3.942255841020	00	00	00	00	00	00
18	1.510989532220	00	00	00	00	00	00
19	51.0089532220	00	00	00	00	00	00
20	1.510989532220	00	00	00	00	00	00
21	3.942255841020	00	00	00	00	00	00
22	3.942255841020	00	00	00	00	00	00
23	3.942255841020	00	00	00	00	00	00
24	1.510989532220	00	00	00	00	00	00
25	51.0089532220	00	00	00	00	00	00
26	1.510989532220	00	00	00	00	00	00
27	3.942255841020	00	00	00	00	00	00
28	3.942255841020	00	00	00	00	00	00
29	3.942255841020	00	00	00	00	00	00
30	1.510989532220	00	00	00	00	00	00
31	51.0089532220	00	00	00	00	00	00
32	1.510989532220	00	00	00	00	00	00
33	3.942255841020	00	00	00	00	00	00
34	3.942255841020	00	00	00	00	00	00
35	3.942255841020	00	00	00	00	00	00
36	1.510989532220	00	00	00	00	00	00
37	51.0089532220	00	00	00	00	00	00
38	1.510989532220	00	00	00	00	00	00
39	3.942255841020	00	00	00	00	00	00
40	3.942255841020	00	00	00	00	00	00
41	3.942255841020	00	00	00	00	00	00
42	1.510989532220	00	00	00	00	00	00
43	51.0089532220	00	00	00	00	00	00
44	1.510989532220	00	00	00	00	00	00
45	3.942255841020	00	00	00	00	00	00
46	3.942255841020	00	00	00	00	00	00
47	3.942255841020	00	00	00	00	00	00
48	1.510989532220	00	00	00	00	00	00
49	51.0089532220	00</					

[illegible]

V.V. MAGN. = 1.007102863340 10

V.V. MAGN. RATE = 1.47648570654D_07

VI. DETAILS OF THE COMPUTER PROGRAM

The detailed derivations of all the equations have been given in Chapters II and III. It is the purpose of this chapter, therefore, merely to facilitate the thorough understanding of how these equations have been implemented in the digital computer program, the FORTRAN listing of which appears in Section C of this chapter.

A. FLOW DIAGRAMS

There are only two areas concerned with the basic computation procedure where the logic becomes at all involved. These are the MAIN program structure itself and the time of flight calculation within the subroutine VECTOR. The other computational subroutines are straightforward procedures for evaluating the equations as derived. The logic is somewhat complicated within the INPUT and PRINT subroutines to provide the very flexible operational features described in Chapter V, Section B. These subroutines, however, are not essential to the basic computational procedure of the program and hence will not be flow diagrammed here. Flow diagrams for the two sections mentioned above (MAIN and VECTOR) are shown in Figs. 16 and 17, respectively, with the equations written in the algebraic notation introduced earlier. The numbers appearing in the left-hand margins of the blocks are external formula numbers and can be correlated directly with the FORTRAN listing of the program in Section C. The titles appearing above some of the blocks correspond with the comments in the listings.

In the MAIN program sketched in Fig. 16, the indicated subroutines are as follows:

<u>Subroutine</u>	<u>Description</u>
INPUT	In conjunction with other subroutines (DINPT, SPACE (LINES), NEWPGE) and with BLOCK DATA, reads in data, performs conversions and initialization calculations, and prints out the input data. Sets ITRAT = 3, KOUNT = 1.
EPHEM	Computes the position and velocity components of two gravitating bodies (in circular orbits) from the known ephemeris time, t_{eph} .
VMASS	Computes the position, magnitude, velocity and magnitude rate of the virtual mass for known positions and velocities of the spacecraft and gravitating bodies of known masses.

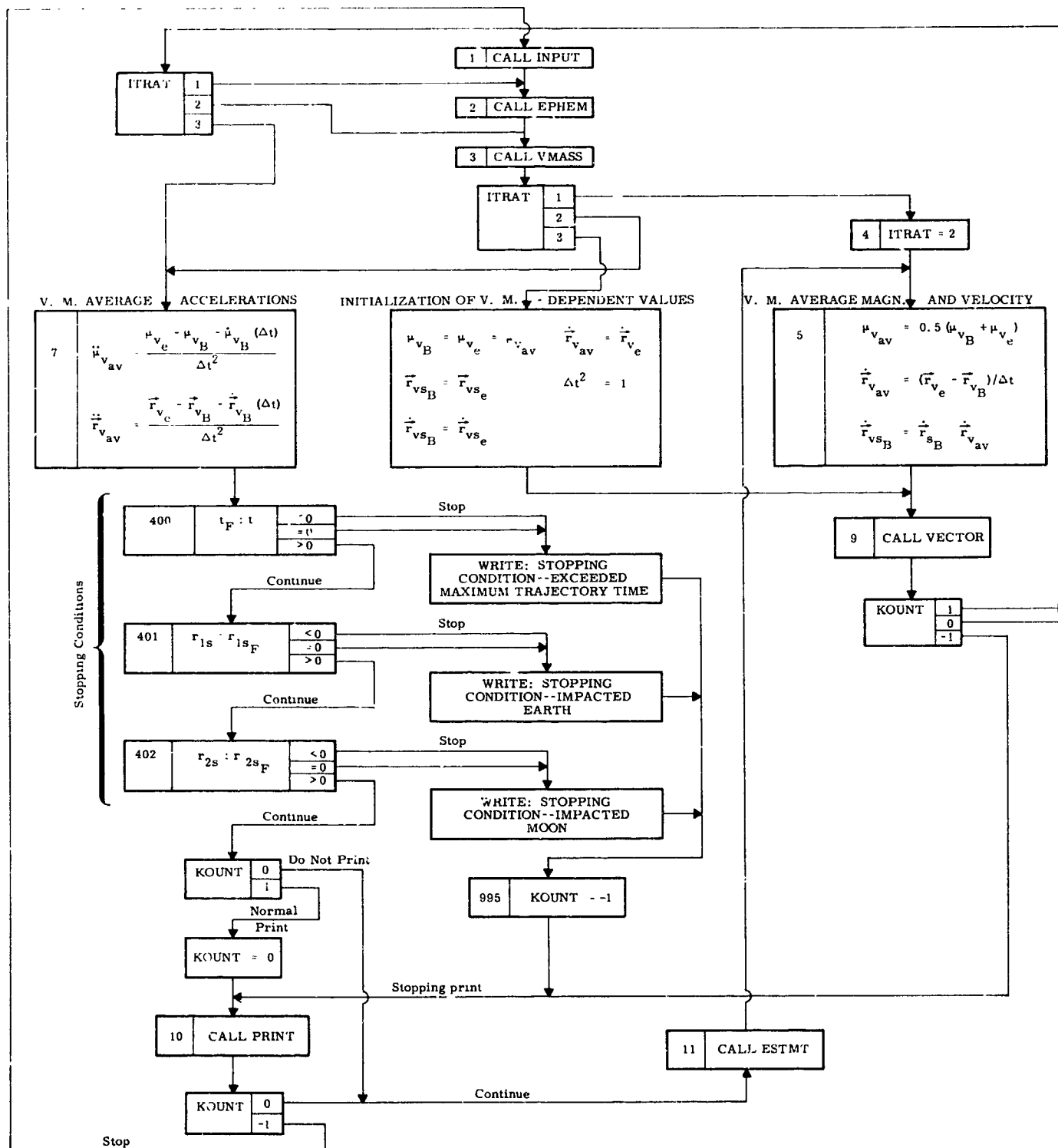


Fig. 16. Flow Diagram of MAIN Program

<u>Subroutine</u>	<u>Description</u>
VECTOR	Calculates the vector orbital elements \vec{k} , \vec{e} , computes the spacecraft final position on the orbit to accurately approximate the desired time interval and then computes the conic section time of flight.
ESTMT	Updates final values of preceding computing interval to serve as initial values for new step (sets ITRAT = 1), determines desired size of time increment on basis of modified true anomaly, major axis crossing or requested print time (sets KOUNT = 1 or 0 depending upon whether regular print is indicated or not), and estimates the final position and magnitude of the virtual mass.
PRINT	In conjunction with subroutines SPACE (LINES) and NEWPGE, performs output conversions and prints out the requested data.

The fixed-point variables ITRAT and KOUNT provide the program logic controls according to:

<u>Variable</u>	<u>Value</u>	<u>Action</u>
ITRAT	1	First pass through computation cycle (including ephemeris)
ITRAT	2	Second and last pass through cycle (excluding ephemeris)
ITRAT	3	Initialization flag
KOUNT	-1	Stopping flag
KOUNT	0	Continue normal computation
KOUNT	1	Print flag

The subroutine VECTOR, shown in Fig. 17, contains two blocks with stars beneath the external formula numbers. This is intended to indicate that the details of the internal logic are not shown for the sake of brevity. In block 510, there are a number of tests to ensure that the argument of the arc sine does not exceed 1. by more than a specified tolerance for the elliptic case. If it does, a stopping condition (KOUNT = -1) is flagged, a return is made to the MAIN program and the logic paths will then terminate the problem. In addition, tests in the listing are not shown for proper quadrant determinations. These tests are straightforward implementations of the procedures described in Chapter III, Section A. Block 520 merely includes some logic to handle the special circumstance where the apocenter for the elliptic case is crossed and the uncorrected equations give a large negative flight time. This, too, is discussed in Chapter III, Section A.

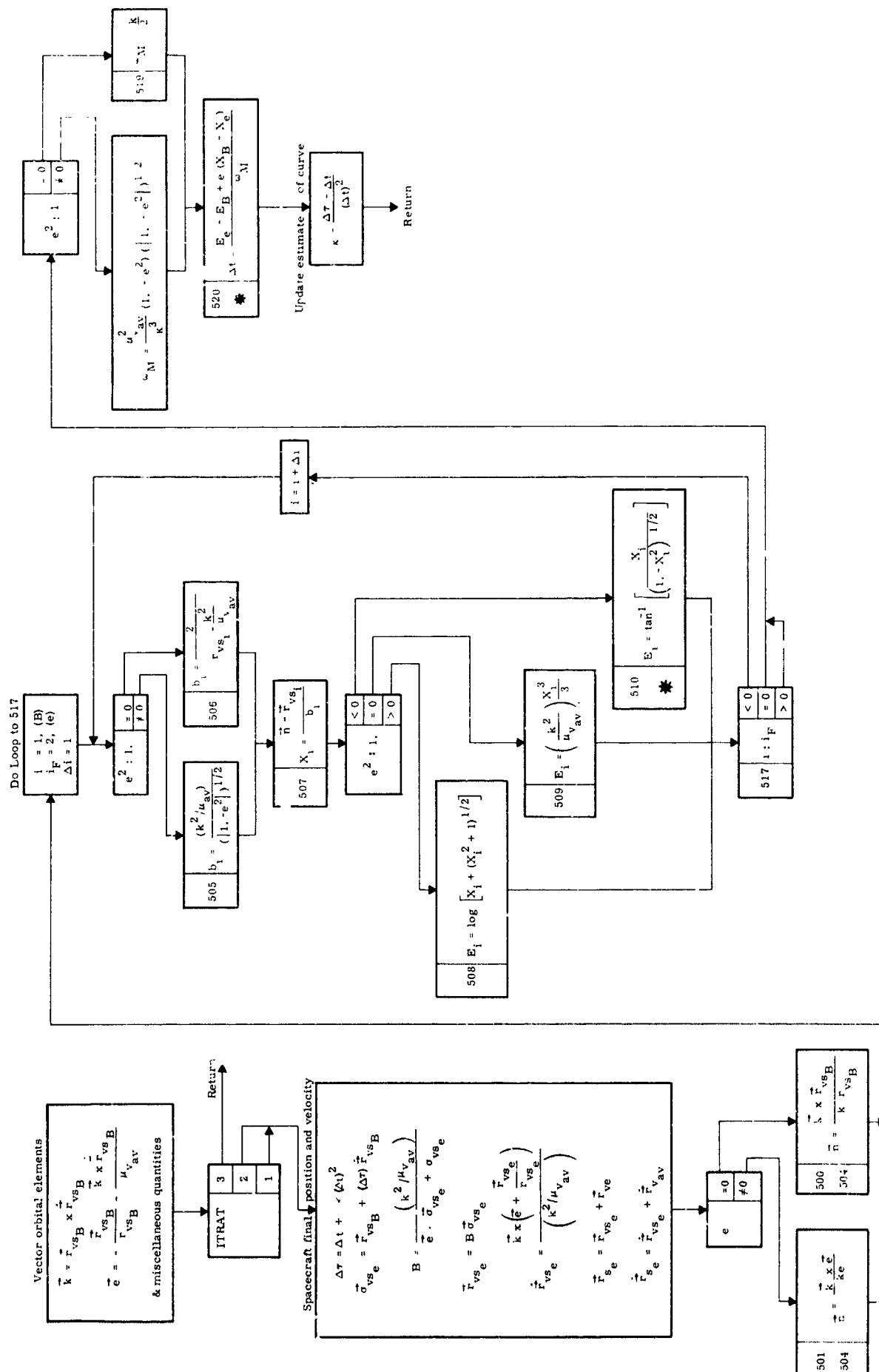


Fig. 17. Flow Diagram of Subroutine Vector

B. ARRAY NOTATION

A glance at the FORTRAN listing in the following section reveals that nearly all the floating point variables have been written in array notation. This makes the job of following the listing all the more difficult. This difficulty is lessened considerably through the aid of Tables I and II in this section. These tables relate the locations in the F (I, J) and V (I, J) arrays to the corresponding algebraic variables--a set of equivalence statements as it were. This slight increase in complexity is deemed well justified on the basis of the conciseness of formulation it affords. With the establishment of appropriate DO loops, all three components of many of the vector expressions can be evaluated by single FORTRAN statements which are essentially identical with the standard vector forms. A testament to the compactness of the basic program is given in the fact that, without the elaborate input and output provisions, the program easily fits in an IBM 1620 computer.

Note that many of the locations in the V (I, J) array find multiple use throughout the program. No attempt has been made to optimize the arrangements and it is quite obvious that more efficient schemes are possible. Such an optimization of the program, though not of immediate interest here, would be required for most efficient machine coding for a computer onboard a spacecraft.

Note further that, although the F (I, J) array appears as an 8 x 4 matrix in Table II for the simple restricted three-body problem, it is dimensioned in the program as F (80, 4). Thus, it can accommodate as many many as twenty gravitating bodies with no change in that part of the program. As mentioned earlier, the subroutine VMASS is also completely general. The DO loops have been established to increment by 4 to the final value

$$\text{NBODY} = 4 * \text{NBODY} - 3$$

where NBODY on the right is stipulated as the number of gravitating bodies. In this program, NBODY = 2 is coded in the INPUT section, but that is the only statement which would have to be changed to consider more than two attractive masses.

TABLE I
V (I, J) Array

I \ J	1	2	3	4	5	6	7
1	$(t_e)_{\text{dim}}, t_B$	$(x_{se})_{\text{dim}}, x_{sB}$	$(y_{se})_{\text{dim}}, y_{sB}$	$(z_{se})_{\text{dim}}, z_{sB}$	ω (deg/t), ω (rad/t)	D	$\mu, -\mu$
2	t_e	x_{se}	y_{se}	z_{se}	$(t_F)_{\text{dim}}, t_F$	$(r_{1s_F})_{\text{dim}}, r_{1s_F}$	$(r_{2s_F})_{\text{dim}}, r_{2s_F}$
3	$(t_{eph_e})_{\text{dim}}, t_{eph_B}$	$(\dot{x}_{se})_{\text{dim}}, \dot{x}_{sB}$	$(\dot{y}_{se})_{\text{dim}}, \dot{y}_{sB}$	$(\dot{z}_{se})_{\text{dim}}, \dot{z}_{sB}$	$(\Delta t_P)_{\text{dim}}, \Delta t_P$	C_2	
4	t_{eph_e}	\dot{x}_{se}	\dot{y}_{se}	\dot{z}_{se}	$\frac{\Delta r}{r}$		ωD (velocity)
5	$(\mu_{ve})_{\text{dim}}, \mu_{vB}$	$(x_{ve})_{\text{dim}}, x_{vB}$	$(y_{ve})_{\text{dim}}, y_{vB}$	$(z_{ve})_{\text{dim}}, z_{vB}$	ωD^2 (area rate)	$\omega^2 D^2$ (velocity) ²	$1 - \mu$
6	μ_{ve}	M_x, x_{ve}	M_y, y_{ve}	M_z, z_{ve}	$\omega^3 D^3$ (mass)	$\omega^3 D^3$ (mass rate)	$\kappa, \Delta \tau$
7	$(\dot{\mu}_{ve})_{\text{dim}}, \dot{\mu}_{vB}$	$(\dot{x}_{ve})_{\text{dim}}, \dot{x}_{vB}$	$(\dot{y}_{ve})_{\text{dim}}, \dot{y}_{vB}$	$(\dot{z}_{ve})_{\text{dim}}, \dot{z}_{vB}$	Δt_k	Δt	$\mu_{v\text{average}}$
8	$\dot{\mu}_{ve}$	M_x, \dot{x}_{ve}	M_y, \dot{y}_{ve}	M_z, \dot{z}_{ve}	$(\Delta t)^2$	$\mu_{v\text{average}}$	B, ω_M
9	r_{vsB}	x_{vsB}	y_{vsB}	z_{vsB}	$x_{vsB}, (\sigma_{vs_e})_x$	$y_{vsB}, (\sigma_{vs_e})_y$	$z_{vsB}, (\sigma_{vs_e})_z$
10	r_{vse}	x_{vse}	y_{vse}	z_{vse}	$\dot{x}_{v\text{avg}}, \ddot{x}_{v\text{avg}}$	$\dot{y}_{v\text{avg}}, \ddot{y}_{v\text{avg}}$	$\dot{z}_{v\text{avg}}, \ddot{z}_{v\text{avg}}$
11	v_{vsB}	x_{vsB}	\dot{y}_{vsB}	\dot{z}_{vsB}	\dot{x}_{vsB}, n_x	\dot{y}_{vsB}, n_y	\dot{z}_{vsB}, n_z
12		$\dot{x}_{vs_e}, e_x + \frac{x_{vs_e}}{r_{vs_e}}$	$y_{vs_e}, e_y + \frac{y_{vs_e}}{r_{vs_e}}$	$z_{vs_e}, e_z + \frac{z_{vs_e}}{r_{vs_e}}$	$M_s, e_x + \frac{x_{vs_e}}{r_{vs_e}}$	$M_s, e_y + \frac{y_{vs_e}}{r_{vs_e}}$	$e_z + \frac{z_{vs_e}}{r_{vs_e}}$
13			t_p	Δt_{MA}	$1 - e_e^2$	$(1 - e_e)^{1/2}$	$\frac{k^2}{\mu_{v\text{avg}}}$
14	e_e	e_{x_e}	e_{y_e}	e_{z_e}	e_e^2, e_{x_e}	$\cos(t_e), e_{y_e}$	$\sin(t_e), e_{z_e}$
15	$(k)_{\text{dim}}$	$(k_x)_{\text{dim}}$	$(k_y)_{\text{dim}}$	$(k_z)_{\text{dim}}$	b_B, X_B	E_B	$a - r_{vs_B}$; or $\vec{r}_{vs_B} \cdot \vec{r}_{vs_B}$
16	k	k_x	k_y	k_z	k_x, k_e^2, b_e, X_e	k_y, E_e	$k, a - r_{vs_e}$; or $\vec{r}_{vs_e} \cdot \vec{r}_{vs_B}$

(1)

TABLE II
F (I, J) Array

I \ J	1	2	3	4
1	x_1	y_1	z_1	μ_1
2	\dot{x}_1	\dot{y}_1	\dot{z}_1	$\frac{3V_{1s}}{2r_{1s}}$
3	x_{1s}	y_{1s}	z_{1s}	r_{1s}
4	\dot{x}_{1s}	\dot{y}_{1s}	\dot{z}_{1s}	$\frac{\mu_1}{3r_{1s}}$
5	x_2	y_2	z_2	μ_2
6	\dot{x}_2	\dot{y}_2	\dot{z}_2	$\frac{3V_{2s}}{2r_{2s}}$
7	x_{2s}	y_{2s}	z_{2s}	V_{2s}
8	\dot{x}_{2s}	\dot{y}_{2s}	\dot{z}_{2s}	$\frac{\mu_2}{3r_{2s}}$

BLANK PAGE

C. FORTRAN IV LISTING OF PROGRAM

\$IBFTC MAIN M94/2,XR7

```

COMMON /COM/ V, F, PI, RAD, VI
DOUBLE PRECISION V(16,7),F(80,4),PI,RAD,VI(16)
COMMON /COM/ ITRAT,KOUNT
COMMON /COM/ NBDYI, NBDY
COMMON /COM/ IPRT(4), IPRTI(4)
COMMON /COM/ KL, IPG, LINCT, LNPGE
COMMON /COM/ NBLOCK
1 CALL INPUT
2 CALL EPHEM
3 CALL VMAS
  IF(ITRAT .EQ. 1) GO TO 4
  IF(ITRAT .EQ. 2) GO TO 7
C INITIALIZATION OF VIRTUAL MASS-DEPENDENT VALUES
  V(7,7) = V(6,1)
  DO 600 J=2,4
    V(10,J+3)=V(8,J)
    V(9,J)=V(10,J)
600 V(11,J)=V(12,J)
    V(9,1)=V(10,1)
    V(8,5)=1.
    V(5,1)=V(6,1)
  GO TO 9
4 ITRAT = 2
C VIRTUAL MASS AVERAGE MAGNITUDE AND VELOCITY
5 V(7,7)=.5*V(5,1)+.5*V(6,1)
  DO 390 J=2,4
    V(10,J+3)=(V(6,J)-V(5,J))/V(7,6)
390 V(11,J)=V(3,J)-V(10,J+3)
9 CALL VECTOR
  IF(KOUNT .LT. 0) GO TO 10
  IF(ITRAT .EQ. 1) GO TO 2
  IF(ITRAT .EQ. 2) GO TO 3
C VIRTUAL MASS AVERAGE ACCELERATIONS
7 V(8,6)=(V(6,1)-V(5,1)-V(7,1)*V(7,6))/V(8,5)
  DO 340 J=2,4
340 V(10,J+3)=(V(6,J)-V(5,J)-V(7,J)*V(7,6))/V(8,5)
C TEST FOR STOPPING CONDITIONS
400 IF (V(2,5) .GT. V(2,1)) GO TO 401
  CALL SPACE (3)
  WRITE (6,4000)
4000 FORMAT (//53H STOPPING CONDITION--EXCEEDED MAXIMUM TRAJECTORY TIME
  $)
  GO TO 995
401 IF (F(3,4) .GT. V(2,6)) GO TO 402
  CALL SPACE (3)

```

```

        WRITE (6,4010)
4010  FORMAT (//35H STOPPING CONDITION--IMPACTED EARTH)
        GO TO 995
402  IF (F(7,4) .GT. V(2,7)) GO TO 403
        CALL SPACE (3)
        WRITE (5,4020)
4020  FORMAT (//35H STOPPING CONDITION--IMPACTED MOON )
        GO TO 995
403  CONTINUE
        IF(KOUNT .EQ. 0) GO TO 11
        KOUNT = 0
10  CALL PRINT
        IF(KOUNT .LT. 0) GO TO 1
11  CALL ESTMT
        GO TO 5
995  KOUNT=-1

```

```

GO TO 10
END

```

\$IBFTC INP M94/2,XR7

SUBROUTINE INPUT

```
COMMON /COM/ V, F, PI, RAD, VI
DOUBLE PRECISION V(16,7),F(80,4),PI,RAD,VI(16)
COMMON /COM/ ITRAT,KOUNT
COMMON /COM/ NBODY1, NBODY
COMMON /COM/ IPRT(4), IPRTI(4)
COMMON /COM/ KL, IPG, LINCT, LNPGE
COMMON /COM/ NBLOCK
COMMON /GCDIN/ ICARD(14)
DIMENSION INCHK(6),FMT(2),CRDTYP(2,6),APRT(4),AAPRT(2,4),IX(3)
DIMENSION NBLK(4)
DOUBLE PRECISION WD(16)
EQUIVALENCE (WPD,WD(1))
DATA INCHK /0,1,1,1,0,1 /
DATA NBLK /10,11,9,12 /
DATA FMT(1) /12H(A24,4D24,0)/
DATA CRDTYP (1,1) /12HPRINT /
DATA CRDTYP (1,2) /12HPOSITION /
DATA CRDTYP (1,3) /12HVELOCITY /
DATA CRDTYP (1,4) /12HEFEMERIS /
DATA CRDTYP (1,5) /12HACCURACY /
DATA CRDTYP (1,6) /12HSTOP /
DATA AAPRT /6H ,6H ,
$ 6H ,6H 1 ,
$ 6H ,6H 2 ,
$ 6H ,6H 3 /
DATA INERR /0/
DATA PRINT / 6HPRINT /
DATA POSITI / 6HPOSITI /
DATA VELOC1 / 6HVELOC1 /
DATA EFEMER / 6HEFEMER /
DATA ACCURA / 6HACCURA /
DATA STOP / 6HSTOP /
DATA PROBLE / 6HPROBLE /
DATA IENTRY /0/
IPG=0
KL=-77777
1 DO 10 I=1,16
DO 10 J=1,7
10 V(I,J)=0.
DO 20 I=1,80
DO 20 J=1,4
20 F(I,J)=0.
V( 3,5)=VI(1)
V( 1,1)=VI(2)
V( 1,2)=VI(3)
V( 1,3)=VI(4)
V( 1,4)=VI(5)
V( 3,2)=VI(6)
```

```

V( 3,3)=VI(7)
V( 3,4)=VI(8)
V( 3,1)=VI(9)
V( 1,5)=VI(10)
V( 1,6)=VI(11)
V( 1,7)=VI(12)
V( 4,5)=VI(13)
V( 2,5)=VI(14)
V( 2,6)=VI(15)
V( 2,7)=VI(16)
DO 30 I=1,4
30 IPRT(I)=IPRT(I)

```

```

100 CALL DINPT (5,NOUT,FMT,WD)
   IF(WRD.EQ.PRINT ) GO TO 110
   IF(WRD.EQ.POSITI) GO TO 120
   IF(WRD.EQ.VELOCI) GO TO 130
   IF(WRD.EQ.EFEMER) GO TO 140
   IF(WRD.EQ.ACCURA) GO TO 150
   IF(WRD.EQ.STOP  ) GO TO 160
   IF(WRD.EQ.PROBLE) GO TO 170
   GO TO 500

```

```

110 CONTINUE
   V(3,5)=WD(2)
   BACKSPACE 5
   READ (5,111) IX

```

```

111 FORMAT (41X3I1)
   DO 113 I=2,4

```

```

113 IPRT(I)=0
   DO 112 I=1,3
   ISUB=IX(I)
   IF (ISUB.EQ.0) GO TO 112
   IPRT(ISUB+1)=1

```

```

112 CONTINUE
   INCHK(1)=0
   GO TO 100

```

```

120 CONTINUE
   V(1,1)=WD(2)
   V(1,2)=WD(3)
   V(1,3)=WD(4)
   V(1,4)=WD(5)
   INCHK(2)=0
   GO TO 100

```

```

130 CONTINUE
   V(3,2)=WD(2)
   V(3,3)=WD(3)
   V(3,4)=WD(4)

```

```

        INCHK(3)=0
        GO TO 100
140  CONTINUE
        V(3,1)=WD(2)
        V(1,5)=WD(3)
        V(1,6)=WD(4)
        V(1,7)=WD(5)
        INCHK(4)=0
        GO TO 100
150  CONTINUE
        V(4,5)=WD(2)
        INCHK(5)=0
        GO TO 100
160  CONTINUE
        V(2,5)=WD(2)
        V(2,6)=WD(3)
        V(2,7)=WD(4)
        INCHK(6)=0
        GO TO 100
170  CONTINUE
        KLT=WD(2)
        GO TO 600
500  IF (INERR.NE.1) CALL NEWPGE
        CALL SPACE (5)
        WRITE (6,501) ICARD
501  FORMAT (/56H AN ERROR HAS BEEN DETECTED READING THE FOLLOWING CARD
        $--/10X,13A6,A2/18H EXECUTION DELETED/1H )
        INERR=1

```

```

        GO TO 100
600  IF (IENTRY.NE.0) GO TO 650
        DO 601 I=1,6
601  IF (INCHK(I).NE.0) GO TO 602
        GO TO 649
602  CALL NEWPGE
        WRITE (6,604)
604  FORMAT (/91H THE FOLLOWING REQUIRED CARD TYPES WERE NOT INPUT ON T
        $HE FIRST PROBLEM -- EXECUTION DELETED)
        DO 605 I=1,6
        IF (INCHK(I).EQ.0) GO TO 605
        WRITE (6,606) (CRDTYP(J,1),J=1,2)
606  FORMAT (10X,2A6)
605  CONTINUE
        INERR=1
649  IENTRY=1
        DO 610 I=1,4
610  IPRTI(I)=IPRT(I)

```

```

VI( 1)=V(3,5)
VI( 2)=V(1,1)
VI( 3)=V(1,2)
VI( 4)=V(1,3)
VI( 5)=V(1,4)
VI( 6)=V(3,2)
VI( 7)=V(3,3)
VI( 8)=V(3,4)
VI( 9)=V(3,1)
VI(10)=V(1,5)
VI(11)=V(1,6)
VI(12)=V(1,7)
VI(13)=V(4,5)
VI(14)=V(2,5)
VI(15)=V(2,6)
VI(16)=V(2,7)
650 IF (INERR.NE.0) GO TO 100
    NBODY=NBODYI
    NBLOCK=0
    DO 660 I=1,4
660 IF (IPRT(I).NE.0) N. LOCK=NBLOCK+NBLK(I)
    CALL NEWPGE
    KL=KLT
    DO 651 I=1,4
    J=IPRT(I)+1
651 APRT(I)=AAPRT(J,I)
    WRITE (6,652) V(3,5),
$           APRT,V(1,1),(V(1,J),J=2,4),(V(3,J),J=2,4),
$           V(3,1),(V(1,J),J=5,7),V(4,5),(V(2,J),J=5,7),KL
652 FORMAT (12H PRINT          1PD20.11,6X4A6/
$          12H POSITION      4D20.11/
$          12H VELOCITY     3D20.11/
$          12H EFEMERIS     4D20.11/
$          12H ACCURACY     D20.11/
$          12H STOP         3D20.11/
$          12H PROBLEM      15)
    CALL NEWPGE
C DIMENSIONAL CONVERSION FACTORS
V(1,5)=V(1,5)/RAD
V(4,7)=V(1,5)*V(1,6)
V(5,5)=V(4,7)*V(1,6)
V(5,6)=V(5,5)*V(1,5)
V(6,5)=V(5,6)*V(1,6)
V(6,6)=V(6,5)*V(1,5)

```

C EPHEMERIS DATA

V(1,7)=-V(1,7)

V(5,7)=1.+V(1,7)

F(1,4)=V(5,7)

F(5,4)=-V(1,7)

NBODY = 2

NBODY=4*NBODY-3

C NONDIMENSIONALIZATION

DO 50 J=2,3

V(J,5)=V(J,5)*V(1,5)

50 V(2,J+4)=V(2,J+4)/V(1,6)

V(2,1)=V(1,1)*V(1,5)

V(4,1)=V(3,1)*V(1,5)

DO 51 J=2,4

V(2,J)=V(1,J)/V(1,6)

51 V(4,J)=V(3,J)/V(4,7)

C INITIALIZATION OF MISCELANEOUS VALUES

V(3,6)=DEXP(1.13756474179255 + .509713741462307*DLGG(V(4,5))

\$ +.14560181279278D-2 * DLOG(V(4,5))**2)

ITRAT = 3

KOUNT = 1

V(13,3)=V(2,1)+V(3,5)

RETURN

END

\$IBFTC EPH XR7,M94/2

SUBROUTINE EPHEM

COMMON /COM/ V, F, PI, RAD, VI

DOUBLE PRECISION V(16,7),F(80,4),PI,RAD,VI(16)

COMMON /COM/ ITRAT,KOUNT

COMMON /COM/ NBDYI, NBDY

COMMON /COM/ IPRT(4), IPRTI(4)

COMMON /COM/ KL, IPG, LINCT, LINPGE

COMMON /COM/ NBLOCK

V(14,7)=DSIN(V(4,1))

V(14,6)=DCOS(V(4,1))

DO 101 I=1,5,4

DO 100 J=1,2

100 F(I,J)=V(1,7)*V(14,J+5)

F(I+1,1)=-F(I,2)

101 F(I+1,2)=F(I,1)

RETURN

END

\$IBFTC VMS M94/2,XR7

SUBROUTINE VMAS

COMMON /COM/ V, F, PI, RAD, VI

DOUBLE PRECISION V(16,7),F(80,4),PI,RAD,VI(16)

COMMON /COM/ ITRAT,KOUNT

COMMON /COM/ NBODYI, NBODY

COMMON /COM/ IPRT(4), IPRTI(4)

COMMON /COM/ KL, IPG, LINCT, LNPGE

COMMON /COM/ NBLOCK

C VIRTUAL MASS POSITION AND MAGNITUDE

V(12,5)=0.

DO 201 I=1,NBODY,4

DO 200 J=1,3

200 F(I+2,J)=V(2,J+1)-F(I,J)

F(I+2,4)=DSQRT(F(I+2,1)**2+F(I+2,2)**2+F(I+2,3)**2)

F(I+3,4)=F(I,4)/F(I+2,4)**3

201 V(12,5)=V(12,5)+F(I+3,4)

DO 203 J=1,3

V(6,J+1)=0.

DO 202 I=1,NBODY,4

202 V(6,J+1)=V(6,J+1)+F(I+3,4)*F(I,J)

V(6,J+1)=V(6,J+1)/V(12,5)

203 V(10,J+1)=V(2,J+1)-V(6,J+1)

V(10,1)=DSQRT(V(10,2)**2+V(10,3)**2+V(10,4)**2)

V(6,1)=V(10,1)**3*V(12,5)

C VIRTUAL MASS VELOCITY AND MAGNITUDE RATE

V(12,6)=0.

DO 301 I=1,NBODY,4

DO 300 J=1,3

300 F(I+3,J)=V(4,J+1)-F(I+1,J)

F(I+1,4)=3.*(F(I+2,1)*F(I+3,1)+F(I+2,2)*F(I+3,2)+F(I+2,3)*F(I+3,3)
1)/F(I+2,4)**2

301 V(12,6)=V(12,6)-F(I+1,4)*F(I+3,4)

DO 303 J=1,3

V(8,J+1)=0.

DO 302 I=1,NBODY,4

302 V(8,J+1)=V(8,J+1)+F(I+3,4)*(F(I+1,J)-F(I,J)*F(I+1,4))

V(8,J+1)=(V(8,J+1)-V(6,J+1)*V(12,6))/V(12,5)

303 V(12,J+1)=V(4,J+1)-V(8,J+1)

V(8,1)=V(6,1)*(3.*(V(10,2)*V(12,2)+V(10,3)*V(12,3)+V(10,4)*V(12,4)
1)/V(10,1)**2+V(12,6)/V(12,5))

RETURN

END

```

$IBFTC VTR      XR7,M94/2
SUBROUTINE VECTOR
COMMON /COM/ V,      F,      PI,      RAD,      VI
DOUBLE PRECISION V(16,7),F(80,4),PI,RAD,VI(16)
COMMON /COM/ ITRAT,KOUNT
COMMON /COM/ NBODYI,  NBODY
COMMON /COM/ IPRT(4), IPRTI(4)
COMMON /COM/ KL,      IPG,      LINCT,      LINPGE
COMMON /COM/ NBLOCK
C VECTOR ORBITAL ELEMENTS
DO 401 J=2,4
DO 400 I=9,11,2
400 V(I,J+3)=V(I,J)
401 V(16,J)=V(9,J+1)*V(11,J+2)-V(9,J+2)*V(11,J+1)
DO 403 J=2,4
DO 402 I=11,16,5
402 V(I,J+3)=V(I,J)
403 V(14,J)=-V(9,J)/V(9,1)-(V(16,J+1)*V(11,J+2)-V(16,J+2)*V(11,J+1))/V
1(7,7)
DO 404 I=14,16,2
V(I,5)=V(I,2)**2+V(I,3)**2+V(I,4)**2
404 V(I,1)=DSQRT(V(I,5))
V(13,5)=1.-V(14,5)
V(13,6)=DSQRT(DABS(V(13,5)))
V(13,7)=V(16,5)/V(7,7)
IF (ITRAT.EQ.3) RETURN
C SPACECRAFT FINAL POSITION AND VELOCITY
V(6,7)=V(7,6)+V(7,6)*V(7,6)*V(6,7)
DO 410 J=2,4
410 V(9,J+3)=V(9,J)+V(6,7)*V(11,J)
V(8,7)=V(13,7)/(V(14,2)*V(9,5)+V(14,3)*V(9,6)+V(14,4)*V(9,7)+DSQRT
1(V(9,5)**2+V(9,6)**2+V(9,7)**2))
DO 411 J=2,4
411 V(10,J)=V(8,7)*V(9,J+3)
V(10,1)=DSQRT(V(10,2)**2+V(10,3)**2+V(10,4)**2)
DO 414 J=2,4
414 V(12,J)=V(14,J)+V(10,J)/V(10,1)
DO 413 J=2,4
DO 412 I=12,16,4
412 V(I,J+3)=V(I,J)
V(12,J)=(V(16,J+1)*V(12,J+2)-V(16,J+2)*V(12,J+1))/V(13,7)
V(2,J)=V(10,J)+V(6,J)
413 V(4,J)=V(12,J)+V(10,J+3)
C KEPLERIAN TIME OF FLIGHT
IF(V(14,1) .NE. 0.) GO TO 501
500 M=9
GO TO 502
501 M=14
502 NN=16-M
DO 504 J=2,4
DO 503 I=M,16,NN

```

```

503 V(I,J+3)=V(I,J)
504 V(11,J+3)=(V(16,J+1)*V(M,J+2)-V(M,J+1)*V(16,J+2))/(V(16,1)*V(M,1))
      DO 517 I=9,10
      IF(V(13,5) .EQ. 0.) GO TO 506
505 V(I+6,5)=V(13,7)/V(13,6)
      GO TO 507
506 V(I+6,5)=2./(V(I,1)-V(13,7))
507 V(I+6,5)=(V(11,5)*V(I,2)+V(11,6)*V(I,3)+V(11,7)*V(I,4))/V(I+6,5)
      IF(V(13,5)) 508,509,510
508 V(I+6,6)=DLOG(V(I+6,5)+DSQRT(V(I+6,5)**2+1.))
      GO TO 517
509 V(I+6,6)=V(14,1)**2*(V(13,7)*V(I+6,5))**3/3.
      GO TO 517
510 IF(DABS(V(I+6,5)) .LT. 1.) GO TO 524
      IF (DABS(V(I+6,5)) .LE. 1.0001) GO TO 597
      KOUNT=-1
      CALL SPACE (2)
      WRITE (6,596)
596 FORMAT (/27H UNACCEPTABLE ERROR IN ATAN)
      RETURN
597 CALL SPACE (2)
      WRITE (6,598)
598 FORMAT (/27H ACCEPTABLE ERROR IN ATAN)
      V(I+6,6)=DSIGN(PI/2.,V(I+6,5))
      GO TO 523
524 V(I+6,6)=DATAN(V(I+6,5)/DSQRT(1.-V(I+6,5)**2))
523 IF(V(14,1) .EQ. 0.) GO TO 512
511 V(I+6,7)=V(I,2)*V(9,2)+V(I,3)*V(9,3)+V(I,4)*V(9,4)
      GO TO 513
512 V(I+6,7)=V(13,7)/V(13,5)-V(I,1)
513 IF(V(I+6,7) .GE. 0.) GO TO 517
      IF(V(I+6,5) .GE. 0.) GO TO 516
      V(I+6,6)=-PI-V(I+6,6)
      GO TO 517
516 V(I+6,6)=PI-V(I+6,6)
517 CONTINUE
      IF(V(13,5) .EQ. 0.) GO TO 519
      V(8,7)=V(7,7)**2/V(16,1)**3*V(13,5)*V(13,6)
      GO TO 520
519 V(8,7)=V(16,1)*V(14,1)**2/2.
520 V(7,5)=V(16,6)-V(15,6)+V(14,1)*(V(15,5)-V(16,5))
      IF(V(7,5) .GE. 0.) GO TO 522
      IF (V(8,7).LT.0.) GO TO 522
      V(7,5)=V(7,5)+2.*PI
522 V(7,5)=V(7,5)/V(8,7)
C UPDATE ESTIMATE OF CURVE
      IF (V(7,5).EQ.0.) GO TO 525
      V(6,7)=(V(6,7)-V(7,5))/V(7,5)**2
525 RETURN
      END

```

318FTC PRT XR7.M94/2

```

SUBROUTINE PRINT
COMMON /COM/ V, F, PI, RAD, VI
DOUBLE PRECISION V(16,7),F(80,4),PI,RAD,VI(16)
COMMON /COM/ ITRAT,KOUNT
COMMON /COM/ NBODYI, NBODY
COMMON /COM/ IPRT(4), IPRTI(4)
COMMON /COM/ KL, IPG, LINCT, LINPGE
COMMON /COM/ NBLOCK
DOUBLE PRECISION TEMP(16),RV,VV,RS,VS
DO 351 J=2,4
DO 350 I=1,5,4
V(I,J)=V(1,6)*V(I+1,J)
V(I+2,J)=V(4,7)*V(I+3,J)
F(I,J-1)=V(1,6)*F(I,J-1)
350 F(I+1,J-1)=V(4,7)*F(I+1,J-1)
351 CONTINUE
DO 352 J=1,4
352 V(15,J)=V(5,5)*V(16,J)
V(1,1)=V(2,1)/V(1,5)
V(3,1)=V(4,1)/V(1,5)
V(5,1)=V(6,1)*V(6,5)
V(7,1)=V(8,1)*V(6,6)
RV=DSQRT(V(5,2)**2+V(5,3)**2+V(5,4)**2)
VV=DSQRT(V(7,2)**2+V(7,3)**2+V(7,4)**2)
RS=DSQRT(V(1,2)**2+V(1,3)**2+V(1,4)**2)
VS=DSQRT(V(3,2)**2+V(3,3)**2+V(3,4)**2)
410 CALL SPACE (NBLOCK)
WRITE (6,411) (V(1,I),I=1,4),RS,(V(3,I),I=2,4),VS
411 FORMAT (//////20H TRAJECTORY TIME = 1PD20.11//
$ 40H SPACECRAFT INERTIAL TRAJECTORY /
$ 40H POSITION . . . . . 4D20.11/
$ 40H VELOCITY . . . . . 4D20.11)
420 IF (IPRT(2).EQ.0) GO TO 430
TEMP( 1) = F(1,1)
TEMP( 2) = F(1,2)
TEMP( 3) = F(1,3)
TEMP( 4) =DSQRT(TEMP(1)**2+TEMP(2)**2+TEMP(3)**2)
TEMP( 5) = F(2,1)
TEMP( 6) = F(2,2)
TEMP( 7) = F(2,3)
TEMP( 8) =DSQRT(TEMP(5)**2+TEMP(6)**2+TEMP(7)**2)
TEMP( 9) = F(5,1)
TEMP(10) = F(5,2)
TEMP(11) = F(5,3)
TEMP(12)=DSQRT(TEMP(9)**2+TEMP(10)**2+TEMP(11)**2)
TEMP(13) = F(6,1)
TEMP(14) = F(6,2)
TEMP(15) = F(6,3)
TEMP(16)=DSQRT(TEMP(13)**2+TEMP(14)**2+TEMP(15)**2)
WRITE (6,421) V(3,1),(TEMP(I),I=1,16)
421 FORMAT (/// 20H EPHEMERIS TIME = 1PD20.11//
$ 40H EPHEMERIS DATA /
$ 40H POSITION OF EARTH . . . . . 4D20.11/

```

```

$          40H      VELOCITY OF EARTH . . . . . 4D20.11//
$          40H      POSITION OF MOON . . . . . 4D20.11/
$          40H      VELOCITY OF MOON . . . . . 4D20.11)

430 IF (IPRT(3).EQ.0) GO TO 440
  TEMP ( 1 ) = V(1,2)-F(1,1)
  TEMP ( 2 ) = V(1,3)-F(1,2)
  TEMP ( 3 ) = V(1,4)-F(1,3)
  TEMP( 4)=DSQRT(TEMP(1)**2+TEMP(2)**2+TEMP(3)**2)

  TEMP ( 5 ) = V(3,2)-F(2,1)
  TEMP ( 6 ) = V(3,3)-F(2,2)
  TEMP ( 7 ) = V(3,4)-F(2,3)
  TEMP( 8)=DSQRT(TEMP(5)**2+TEMP(6)**2+TEMP(7)**2)
  TEMP ( 9 ) = V(1,2)-F(5,1)
  TEMP (10) = V(1,3)-F(5,2)
  TEMP (11) = V(1,4)-F(5,3)
  TEMP(12)=DSQRT(TEMP(9)**2+TEMP(10)**2+TEMP(11)**2)
  TEMP (13) = V(3,2)-F(6,1)
  TEMP (14) = V(3,3)-F(6,2)
  TEMP (15) = V(3,4)-F(6,3)
  TEMP(16)=DSQRT(TEMP(13)**2+TEMP(14)**2+TEMP(15)**2)
  WRITE (6,431) (TEMP(I),I=1,16)

431 FORMAT(/// 40H SPACECRAFT RELATIVE TRAJECTORIES /
$          40H      POSITION REL. TO EARTH . . . . . 4D20.11/
$          40H      VELOCITY REL. TO EARTH . . . . . 4D20.11//
$          40H      POSITION REL. TO MOON . . . . . 4D20.11/
$          40H      VELOCITY REL. TO MOON . . . . . 4D20.11)

440 IF (IPRT(4).EQ.0) RETURN
  TEMP(1)=V( 9,2)*V(1,6)
  TEMP(2)=V( 9,3)*V(1,6)
  TEMP(3)=V( 9,4)*V(1,6)
  TEMP(4)=V( 9,1)*V(1,6)
  TEMP(5)=V(11,2)*V(4,7)
  TEMP(6)=V(11,3)*V(4,7)
  TEMP(7)=V(11,4)*V(4,7)
  TEMP(8)=DSQRT(TEMP(5)**2+TEMP(6)**2+TEMP(7)**2)
  WRITE (6,441) (V(5,I),I=2,4),RV,(V(7,I),I=2,4),VV,
$      (TEMP(I),I=1,8),
$      (V(15,I),I=2,4),V(15,1),(V(14,I),I=2,4),V(14,1),
$      V(5,1),V(7,1)

441 FORMAT(/// 20H VIRTUAL MASS DATA /
$          40H      VIRTUAL MASS POSITION . . . . . 4(IPD20.11
$)/          40H      VIRTUAL MASS VELOCITY . . . . . 4D20.11/
$          40H      SPACECRAFT POS. REL. TO V.M. . . . . 4D20.11/
$          40H      SPACECRAFT VEL. REL. TO V.M. . . . . 4D20.11/
$          40H      KEPLER (ANG. MOM.) VECTOR. . . . . 4D20.11/
$          40H      ECCENTRICITY VECTOR . . . . . 4D20.11/
$          24H      V.M. MAGN. = D20.11/
$          24H      V.M. MAGN. RATE = D20.11)

  RETURN
  END

```

```

$IBFTC EST      XR7,M94/2
      SUBROUTINE ESTMT
      COMMON /COM/ V,      F,      PI,      RAD,      VI
      DOUBLE PRECISION V(16,7),F(80,4),PI,RAD,VI(16)
      COMMON /COM/ ITRAT,KOUNT
      COMMON /COM/ NBODYI,  NBODY
      COMMON /COM/ IPRT(4), IPRTI(4)
      COMMON /COM/ KL,      IPG,      LINCT,      LNPGE
      COMMON /COM/ NBLOCK
C INDEX VALUES IN V ARRAY
      DO 361 I=1,9,2
      DO 360 J=1,4
360 V(I,J)=V(I+1,J)
361 CONTINUE
      ITRAT=1
C ESTABLISH COMPUTING TIME INCREMENT
395 V(13,4)=V(7,6)*V(16,5)/(V(15,5)-V(16,5))
      IF (V(7,6).EQ.0.) V(13,4)=-1.
      V(11,1)=DSQRT(V(11,2)**2+V(11,3)**2+V(11,4)**2)
      V(7,6)=V(3,6)*V(9,1)/V(11,1)
      IF(V(13,4) .LT. 0.) GO TO 394
      IF(V(13,4) .GT. 1.1*V(7,6)) GO TO 394
      V(7,6)=V(13,4)
      CALL SPACE (5)
      WRITE (6,6000)
6000 FORMAT (//2X,19HMAJOR AXIS CROSSING//1H )
      GO TO 400
394 IF(V(1,1)+1.1*V(7,6) .LT. V(13,3)) GO TO 378
390 V(7,6)=V(13,3)-V(1,1)
      V(13,3)=V(13,3)+V(3,5)
400 KOUNT = 1
C INCREMENT TIMES
378 DO 379 I=1,3,2
379 V(I+1,1)=V(I,1)+V(7,6)
      V(8,5)=V(7,6)**2
      IF (V(2,1).GE.V(13,3)) V(13,3)=V(13,3)+V(3,5)
C ESTIMATE VIRTUAL MASS FINAL POSITION AND MAGNITUDE
      V(6,1)=V(5,1)+V(7,1)*V(7,6)+V(8,6)*V(8,5)
      DO 380 J=2,4
380 V(6,J)=V(5,J)+V(7,J)*V(7,6)+V(10,J+3)*V(8,5)
      RETURN
      END

```

\$IBFTC BLKDAT XR7,M94/2

BLOCK DATA

COMMON /COM/ V, F, PI, RAD, VI

DOUBLE PRECISION V(16,7),F(80,4),PI,RAD,VI(16)

COMMON /COM/ ITRAT,KOUNT

COMMON /COM/ NBODYI, NBODY

COMMON /COM/ IPRT(4), IPRTI(4)

COMMON /COM/ KL, IPG, LINCT, LINPGE

COMMON /COM/ NBLOCK

DATA PI /3.141592653589793 /

DATA RAD /57.29577951308232 /

DATA NBODYI / 2 /

DATA IPRTI / 1,0,0,0 /

DATA LINPGE/60/

DATA VI / 5.D0, 11*0.D0, 1.D-7, 3*0.D0 /

END


```
$IBFTC DIN      XR7,M94/2
  SUBROUTINE DINPT (X,XX,XXX,WD)
  DOUBLE PRECISION WD(5)
  COMMON/GCDIN/ ICARD(14)
  READ (5,1) ICARD
1  FORMAT (13A6,A2)
  BACKSPACE 5
  READ (5,2) WD
2  FORMAT (A6,2X,4D18.0)
  RETURN
END
```

```

$IBFTC SPC      XR7,M94/2
SUBROUTINE SPACE (LINES)
COMMON /COM/ V,      F,      PI,      RAD,      VI
DOUBLE PRECISION V(16,7),F(80,4),PI,RAD,VI(16)
COMMON /COM/ ITRAT,KOUNT
COMMON /COM/ NBOYI,  NBOY
COMMON /COM/ IPRT(4), IPRTI(4)
COMMON /COM/ KL,      IPG,      LINCT,      LIMPGE
COMMON /COM/ NBLOCK
IF (LIMPGE.LT.(LINCT+LINES)) CALL NEWPGE
LINCT=LINCT+LINES
RETURN
END

```

```

$IBFTC NPG      XR7,M94/2
SUBROUTINE NEWPGE
COMMON /COM/ V,      F,      PI,      RAD,      VI
DOUBLE PRECISION V(16,7),F(80,4),PI,RAD,VI(16)
COMMON /COM/ ITRAT,KOUNT
COMMON /COM/ NBODYI,  NBODY
COMMON /COM/ IPRT(4), IPRTI(4)
COMMON /COM/ KL,      IPG,      LINCT,      LNPGE
COMMON /COM/ NBLOCK
IPG=IPG+1
WRITE (6,1)
1 FORMAT (120H
$1  V I R T U A L      M A S S      P R O G R A M      F O R      C O M P
$U T I N G      S P A C E      T R A J E C T O R I E S      )

C
C      WHEN KL = -77777, ONLY TITLE AND PAGE NUMBER ARE GIVEN, AS
C      THIS SIGNALS INPUT DIAGNOSTICS ARE TO BE GIVEN, OR INPUT DATA
C      IS TO BE LISTED.
C

      IF (KL.EQ.-77777) G. TO 10
      WRITE (6,2) KL,IPG
2 FORMAT (90X8HPROBLEM 15,6X5HPAGE 14///
$      40X80H      X - COMP.
$      Y - COMP.      Z - COMP.      RESULTANT )
      LINCT=6
      RETURN
10 WRITE (6,3) IPG
3 FORMAT (109X5HPAGE 14)
      LINCT=2
      RETURN
END

```

VII. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

The results of this study have shown that the virtual mass technique provides a practicable and very flexible method for solving the n-body problem. Identically the same computer program can be controlled by a single input to obtain approximate solutions very quickly, or highly accurate trajectories in a proportionately longer time. The sample circumlunar trajectory included in this report gives the spacecraft position accurate to within 0.02 naut mi at $t = 70$ hr (approximately 0.33 hr before pericynthion) in 160 sec on an IBM 7094 digital computer.

The use of rectangular coordinates and the formulation of the conic section relationships in terms of the vector orbital elements \vec{k} , \vec{e} have resulted in a computationally compact program. Without the elaborate input-output provisions which have been incorporated to provide operational flexibility, the basic computational program easily fits in an IBM 1620 computer.

B. RECOMMENDED FURTHER STUDIES

A number of further studies are suggested and will be listed here without elaboration.

- (1) Derive analytical expressions for trajectory sensitivities from the simple conic section forms relative to the virtual mass. Use these to propagate the state transition matrix analytically.
- (2) Study techniques for representing aspherical gravitational potentials by appropriate planet-fixed distributions of discrete point masses similar to the method of Ref. 4. Investigate the integration of such trajectories by the virtual mass technique.
- (3) Study the problem of computing dynamically consistent trajectories (see Refs. 2 and 5) by investigating an extension of the virtual mass technique to compute the simultaneous trajectories followed by n gravitating bodies. The procedure would be to reduce the problem to a series of n two-body systems at every instant. Each two-body system would consist of a different one of the n real bodies and a corresponding fictitious body lumping the effects of all others on the one of immediate interest. The numerical computation accuracy would be controlled so as to conserve known integrals: energy, momentum and uniformity of motion of the center of mass.

- (4) Develop an Encke-like procedure for computing low thrust trajectories. Here the thrust "perturbation" would be integrated separately and added as a correction to the reference gravitational trajectory relative to the virtual mass.
- (5) Perform a general study of trajectories of the virtual mass to ascertain, if possible, the fundamental characteristics of its motion. Also try to find an analytical solution to the variable mass two-body problem.

VIII. REFERENCES

1. Gartmann, H. "Raumfahrtforschung." R. Oldenbourg, Munchen, 1952.
2. Novak, D. "Two-Body Midcourse Navigation and Guidance for Lunar Missions." AIAA Preprint No. 64-649, Astrodynamics Guidance and Control Conference, Los Angeles, August 1964.
3. Miner, W. and Sperling, H. "The Varicentric Method of Space Flight Calculations with Introduction of the Ephemerides." MSFC Report No. DA-TM-93-59, December 1959.
4. Vinti, John P. "Theory of an Accurate Intermediary Orbit for Satellite Astronomy." Journal of Research of the National Bureau of Standards, B. Mathematics and Mathematical Physics, Vol. 65B, No. 3, July-September, 1961.
5. "Explanatory Supplement to the Ephemeris." Her Majesty's Stationery Office, London, 1961.